AUTOMATING ECLIPSES
If you have a problem that can be solved by a computer—we have a systems solution.

- Two central processors with maximum RAM capacities of 56K and 384 K bytes
- Three types of disk drives with capacities of 175K, 1.2M and 16M bytes
- Two dot matrix printers with 80 and 132 line capacity
- A Selectric typewriter interface and a daisy wheel printer

Match these to your exact need, add one or more of our intelligent terminals and put together a system from one source with guaranteed compatibility in both software and hardware.

Southwest Technical Products systems give you unmatched power, speed and versatility. They are packaged in custom designed woodgrain finished cabinets. Factory service and support on the entire system and local service is available in many cities.
You can do surprising things when you have 64 kilobytes of fast RAM on one card

4 MHz FAST—AND EXPANDABLE
Here's 64 kilobytes of memory on one RAM card. Yes, we mean 512K bits of read/write memory on this single card.
And, yes, we mean it's fast. With 150-nanosecond chip access times — so the card can operate in fast Z-80 systems with no wait states. Repeat, no wait states.

EXPANDABLE ON TWO LEVELS
Not only does the new Model 64KZ give you a large, fast RAM but it is expandable on two levels.
First, through our Cromemco Bank Select feature, you can expand to 512 kilobytes in eight 64K banks.
Or, with our Extended Bank Select feature, you can expand memory space to as much as 16 megabytes. This expandability we call your obsolescence insurance.
The legend on the card's heat sink is an easy reference for address and bank selection.

BENCHMARK IT
Obviously, the speed and memory capacity of this new card give you a lot of power.
You can see that for yourself in our new 7-station Multi-User Computer System which uses these Model 64KZ cards. This S100-bus system outperforms the speed of many if not most timesharing systems of up to 10 times the Cromemco price.
And yet where some of these much more expensive and cumbersome systems clearly slow to a snail's pace when timesharing, the Cromemco system using Bank Select switching runs surprisingly fast.

SEE IT NOW
See the new Model 64KZ at your computer dealer now. Study the literature on it. See how for only $1785 you can get around that ever-present barrier of memory that's too little and too slow.

Cromemco

Circle 80 on inquiry card.
Here's how you can be fully computerized for so much less than you thought

BUSINESS — EDUCATION — ENGINEERING — MANUFACTURING

We are pleased to announce the first professional time-sharing system in the microcomputer field.

Naturally, it's from Cromemco. This new multi-user system will do all of the tasks you usually associate with much more expensive time-sharing computers. Yet it's priced at an almost unbelievably low figure.

Look at these features:
• You can have up to 7 terminals plus a fast, 132-column line printer
• You can have a large system RAM memory that's expandable to ½ megabyte using the Bank Select feature
• Each user has an independent bank of RAM
• You can have floppy disk storage of up to 1 megabyte
• You have confidentiality between most stations
• And, make no mistake, the system is fast and powerful. You'll want to try its fast execution time yourself.

PROGRAMMERS LOVE OUR BASIC

This new system is based on Cromemco's well-known System Three Computer and our new Multi-User BASIC software package.

Programmers tell us that Cromemco Multi-User BASIC is the best in the field. Here are some of its attractions:
• You can use long variable names and labels up to 31 characters long — names like "material on order" or "calculate speed reduction."
• You get many unusual and helpful commands that simplify programs and execution — commands such as PROTECT, LIST VARIABLES, NOLIST, and many more.
• No round-off error in financial work (because our BASIC uses binary-coded decimal rather than binary operation). And we've still been able to make it FAST.
• Terminals and printer are interrupt-driven — no additional overhead until key is pressed.
• The conveniences in this Multi-User BASIC make it much easier to write your own application software.
• A line editor simplifies changes. BENCHMARK IT — NOW

In the final analysis, the thing to do is see this beautiful new system at your dealer. See its rugged professional quality. Evaluate it. Benchmark it for speed with your own routine (you'll be agreeably surprised, we guarantee you). Find out, too, about Cromemco's reputation for quality and engineering.

Look into it now because you can have the capabilities of a fully computerized operation much quicker and for much less than you ever thought.
In The Queue

Foreground

18 GRAPHIC INPUT OF WEATHER DATA, by Stephen P Smith
Graphical data acquisition techniques

34 SOUND OFF, by Steve Ciarcia
Interfacing a computer to external sound generation devices

54 A MODEL OF THE BRAIN FOR ROBOT CONTROL, Part 2: A Neurological Model, by James Albus
A hierarchical network that can execute tasks and seek goals

120 SONIC ANEMOMETRY FOR THE HOBBYIST by Neil Dvorak
Measuring wind speed and direction by electronic means

Understanding closed loop control systems

176 QUEST, by Roger Chaffee
Will you find the treasure or will the giant find you first?

198 MOUSE, A Language for Microcomputers, by Peter Grogono
Simple language to demonstrate implementation processes

226 SUBROUTINE PARAMETERS, by W D Maurer
More ways than you ever dreamed possible to give your subroutines the data they need

Background

105 THE MATHEMATICS OF COMPUTER ART, by Kurt Schmucker
Plotting figures that express mathematical relationships

158 CREATIVITY IN COMPUTER MUSIC, by Hubert S Howe Jr
Effect of computer use on music composition and theory

194 PHOTO ESSAY: Physical Hardware of a New Computer Backplane, by Carl T Helmers Jr
The beginnings of a homebrew 6809 personal computer

Nucleus

6 Letters

8 Editorial, "Computers and Eclipses"

36,192 Programming Quickies

96 Event Queue

98,156 BYTE's Bugs

99 BYTE News

154 Clubs and Newsletters

155 BYTE's Bits

174 Nybbles: Tiny Pascal in Assembly Language

192 World Power Systems: A Report

187 Book Reviews

222 Technical Forum

231 Languages Forum

233 What's New?

270 Unclassified Ads

272 BOMB, Reader Service

Cover Art: AUTOMATING ECLIPSES, by Robert Timney

BYTE is published monthly by BYTE Publications Inc, 70 Main St, Peterborough NH 03458. A wholly-owned subsidiary of McGraw-Hill, Inc. Address all mail except subscriptions to above address: phone (603) 924-7217. Address subscriptions, change of address, USPS Form 3679, and fulfillment questions 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 (ISSN 0360-5380). Subscription rates are $18 for one year, $32 for two years, and $48 for three years in the USA and its possessions. In Canada and Mexico, $20 for one year, $36 for two years, $52 for three years. $32 for one year air delivery to Europe, $22 surface delivery elsewhere. Air delivery to selected areas at additional rates upon request. Single copy price is $2 in the USA and its possessions, $2.40 in Canada and Mexico, $3.50 in Europe, and $4 elsewhere. Foreign subscriptions and sales should be remitted in United States funds drawn on a US bank. Printed in United States of America. 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. Entire contents copyright © 1979 by BYTE Publications Inc. All rights reserved.

BYTE® is available in microform from University Microfilms International, 300 N Zeeb Rd, Dept PR, Ann Arbor MI 48106 USA or 18 Bedford Row, Dept PR, London WC1R 4EJ ENGLAND.

Subscription WATS Line: (800) 258-5485 Office hours: Mon-Thur 8:30 AM – 4:30 PM Friday 8:30 AM – Noon

In This BYTE

When entering large amounts of graph associated data into a computer, a graphic tablet that allows you to digitize the data is a great help. Stephen P. Smith uses such a device, the Summagraphics Bit Pad digitizer, to perform the Graphic Input of Weather Data. page 16

This month "Ciarcia's Circuit Cellar" explores the use and interfacing of Texas Instruments and General Instrument sound generators. Find out how you can let your computer Sound Off. page 34

In part 1 of A Model of the Brain for Robot Control, James Albus defined the notation we used for his brain model. This month he describes a neurological model that can store and recall a broad class of mathematical functions. page 54

Much computer art employs the calculating ability of the machine to make drawings expressing mathematical relationships. Kurt Schmucker defines two classes of such drawings and describes methods for producing them in The Mathematics of Computer Art. page 105

To forecast weather, you need to know wind speed and direction. By using modern technology, we can do without whirling mechanical assemblies. Neil Dvorak shows us how to use electronic components and computer programs to measure the wind in Sonic Anemometry for the Hobbyist. page 120

In part 2 of The Nature of Robots, William T. Powers presents a BASIC simulation of a control system. By experimenting with this simulator, the reader is able to work with the concepts of a closed loop control system. page 134

Creativity in Computer Music by Hubert S. Howe Jr. is a survey of some recent work in music theory, analysis, sound generation, and composition done with computers. Microcomputers can now be used for much of the work formerly done by large scale computers a decade ago. page 158

After you have successfully hunted the Wumpus, and destroyed all the Klingons, what is your next step? Roger Chaffee suggests you try your hand in some caves, searching for hidden treasure. Enter the world of suspense and danger on your Quest for riches. page 176

Building a computer from scratch as an amateur is the historical root of the personal computer field. In this issue, Carl Helmers begins an informal series of articles on a new homebrew project: a general purpose 6809 system. The computer itself has an intended application to music, but the design and construction of this homebrew project are quite general. See Photo Essay: Physical Hardware of a New Computer Backplane. page 194

Mouse is a programming language that contains many features usually associated with high level programming languages and can be implemented with minimal resources. It is of interest to people who enjoy obtaining dramatic results with little effort and to those who have a system which is too small to support a conventional high level language. Peter Grogono describes the implementation of Mouse by means of a Pascal program which can be used to write an assembly language version. Indications of how this might be done are provided in Mouse: A Language for Microcomputers. page 198

When working with subroutines, the concept of passing parameters can be confusing. W. D. Maurer describes three methods of passing parameters (call by value and result, call by reference, and call by name) in his article Subroutine Parameters. page 226
Good News for Smart People

TDS-IB Disk System

HITACHI BASIC MASTER Compatible

Coming Soon:
- TDS-IA
  Apple II
  Compatible
- TDS-IT
  TRS-80
  Compatible
- 35 Tracks with ten 256 byte sectors per Track.
- 87.5K bytes per disk.
- Interface, power-unit included.
- Controls up to four drives.
- Software included. TDOS (TIP Disk Operating System).
- OPTION: Business utility programs.

TRADE OF INDUSTRIAL PRODUCTS INC.

2-1-19-815 Kanda-Surugadai
Chiyoda-ku, Tokyo 101, Japan
(03) 295-7055
TELEX 02226152 TIPINC J

Circle 370 on inquiry card.
SERIOUS COMMENTS ON AMENDED BASIC

I enjoyed the humor that Robert Bass used in his Languages Forum on "Amended BASIC" (April 1979 BYTE, page 238). Most of his suggested additions to the BASIC language were of the April Fool's Day variety; however, his FORGET statement, though included in jest, does have some merit. Frequently, the programs I write have hard-to-find bugs in them. These programs may have long printouts of instructions on how to use them. However, when debugging programs, it is irritating to have to wait for all of these printouts before the program really starts. I usually change all of those PRINT statements into comments by inserting a REM before them. A FORGET statement, however, would be far more convenient. The programmer could include a statement like FORGET 100-210, 320, 400-460 at the beginning, and then run the program. The BASIC will treat the statements listed in the FORGET statement as remarks. When you are finished debugging the program, remove the FORGET statement, or change it into a remark. In this way, you need to change only one statement in your program, instead of numerous statements as in our current BASIC.

Another suggestion I would like to see implemented in BASIC is a variation of the RESTORE statement. Presently, the RESTORE statement sets the DATA pointer to the start of the list of DATA. However, sometimes it is convenient to have the pointer set at a different point. I suggest that a statement of the form "RESTORE 300" be implemented. This would set the DATA pointer to the first set of DATA at or following line 300. A variation of this might be "ON K RESTORE 300, 310,320." This statement would be analogous to the "ON K GOTO 300, 400, 500" statement. Both versions would allow immediate access to DATA. At present, you must RESTORE the pointer to the beginning of the DATA, and then use dummy variables to READ the DATA you actually want.

James L. Bostler
Director of the Computer Laboratory
Ciafin College
Orangeburg, SC 29115

SQUISH BUGS

Regarding the April 1979 editorial about operating systems with bombed file systems:
First, we note that the UCSD file system, RT-11 for the PDP-11, and many other disk operating systems require periodic squishes to manage a disk. This is a foolhardy stunt as, in your case, one bad disk sector can prevent the entire squash from working. I feel that people who build and propagate file systems like this (without even any attempt to skip bad sector) are irresponsible.

It shows the need for better error recovery or a scheme which prevents the need for squish altogether, such as dynamic file space allocations (eg: CP/M).

Secondly, your need to write your own recovery program indicated a need for such a recovery program to become as a standard operating system utility. Disasters happen; the need is real.

To my knowledge, Motorola MDOS and Software Dynamic's SDOS (for the 6800) are the only microcomputer operating systems that provide both dynamic file allocation and disaster recovery programs.

The industry needs more systems like these.

Ira Baxter
Software Dynamics
2111 W Crespien, Suite G
Anaheim, CA 92801

A FASTER MAILING LIST

In reference to Thomas E. Doyle's article, "A Computerized Mailing List," (January 1979 BYTE, page 84) a few modifications might be helpful, particularly, as he expresses some concern on saving time in the discussion of Program 6.

Program 7 is the main concern of this discussion. After the program locates the desired record to eliminate, it performs, in closing up the gap, what is commonly called garbage collection. It would appear to be more desirable to flag that record as an unused record and not perform garbage collection every time. The ways to do this are: a special (in Mr. Dove's application) call sign which is recognized as a null record; or add an additional variable to the list for each record. This additional variable could be easily used to indicate a variety of meanings for the remaining data on the record besides a null record. Then Programs 2 and 3 could be selective.

With the addition of this variable, all the other programs would have to be modified to take into account the change in the structure of the file. The disk file could be viewed as a collection of one or more sets of records, each set being zero or more consecutive records of good data, and ending with one null record. Program 1, having initialized the file, would then write one null record before ending. Program 6 would only search up to the first null record after the point of insertion (a null record must still exist at the end of the file, for Program 4 as well). Program 8 (to be

Text continued on page 98
“After working all day with the computer at work, it's a kick to get down to Basic at home. And one thing that makes it more fun is my Shugart minifloppy™. We use Shugart drives at work, so when I bought my own system I made sure it had a minifloppy drive.

"Why? Shugart invented the minifloppy. The guys who designed our system at work tell me that Shugart is the leader in floppy design and has more drives in use than any other manufacturer. If Shugart drives are reliable enough for hard-working business computers, they've got to be a good value for my home system.

"When I'm working on my programs late at night, I can't wait for cassette storage. My minifloppy gives me fast random access and data transfer. The little minidiskettes™ store plenty of data and file easily too.

"I made the right decision when I bought a system with the minifloppy. When you lay out your own hard-earned cash, you want reliability and performance. Do what I did. Get a system with the minifloppy."

If it isn't Shugart, it isn't minifloppy.

Shugart
435 Oakmead Parkway, Sunnyvale, California 94086

See opposite page for list of manufacturers featuring Shugart's minifloppy in their systems.

™ minifloppy is a registered trademark of Shugart Associates
Computers and Eclipses

by Carl Helmers

The idea occurred last fall. An innocuous advertisement appeared in *Smithsonian Magazine,* trumpeting an exciting adventure called "Eclipse Over Big Sky" which would take place in February 1979 at the Big Sky ski resort in the southwestern part of Montana, near Yellowstone Park in Wyoming. Naturally, I sent away for the information advertised.

After receiving the literature, I made up my mind that a total solar eclipse was worth seeing, especially if it was to be the last one on continental North America for some forty years. So, I sent in my deposit and made plans to attend. It turned out (as I found when I arrived) that this expedition was one of a series of such expeditions organized by sociologist and eclipse buff Dr Phil Sigler of New York City. These expeditions had attended every total solar eclipse for the past eight or ten years. Using the latest in modern techniques, including reference to weather satellite data, they had found a necessary hole in the clouds at the right time in eight out of nine cases prior to this eclipse.

Previous expeditions had used cruise ships on the open ocean in order to implement the concept of "mobility" pioneered by Dr Ed Brooks of Boston University, the weather adviser for the operation. In order to utilize the same concept for the 1979 eclipse, some form of land mobility was required. An initial attempt to take advantage of an Amtrak route which paralleled the eclipse path was apparently squelched by the usual bureaucratic catch-22: "Sure you can rent the track, but we can't supply you with a train." Thus, mobility was achieved through the services of the Yellowstone Bus Company and a procession of 15 large buses. At 2 AM on eclipse day, this procession left the hotel for a six hour trip to central Montana, just west of a town called Roundup.

Taking pictures of a transient, two minute phenomenon is one of the goals of an eclipse expedition; the other goal being to simply watch this phenomenon with the naked eye or through a suitable telescope. When I say "naked eye," I mean it, despite all normal reactions which say "you can't look at an eclipse without protection." In actuality, there is absolutely no way to look at an un-eclipsed or partially eclipsed sun without using filters to avoid damaging your eyeballs or camera equipment.

However, this is the key difference with regard to a totally eclipsed sun: you can look at it directly. That last .1 percent that separates 99.9 from 100 percent makes all the difference in the world between the dull, filtered crescent sun of a partial eclipse and the incredibly beautiful natural phenomenon of a totally eclipsed sun. You can take excellent pictures, without filters, using 400 speed film and exposures of 1/30 to 1/2000 of a second (see photos 1 and 2). But, photographing the phenomenon is definitely a bit of a problem.

Durings this past eclipse, I had only enough time to take about 15 exposures, with one lens change. This was done in -3 to 0 degree Celsius prevailing temperatures on an isolated road west of Roundup, Montana. My hands froze, and I probably did not get the optimal personal viewing, although the 1000 mm reflex telephoto lens of my camera acted as an excellent spotting scope through which to watch the sun for most of the eclipse.
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.

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.

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.

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.
The problem is that if you spend your allotted time budget fooling around with the camera, you can miss a good portion of the event and its natural beauty. This is where the computer experimenter's inventiveness can come into play. Why not automate the exposure and picture taking sequences of the camera and telescope combination, so that once the first diamond rings of totality occur, a microcomputer can run through an open loop exposure sequence adapted to the camera equipment and the particular eclipse being viewed?

What are the functional requirements of such a device? Based on the recommendations of the expedition's photography advisers, George Keene of Eastman Kodak and Robert Little of Criterion Manufacturing (and confirmed by my own successful experience), the main requirement for achieving excellent photos with a 35 mm camera during an eclipse is to use a non-automatic exposure technique which simply covers a range of shutter speed settings within a fixed aperture setting.

At each exposure of the film, different phenomena dominate the image. During short exposures, the extremely bright solar prominences are highlighted, with almost no corona visible (see photo 1). In longer exposures, one begins to see details of the fainter solar corona, while the inner prominence detail washes out due to overexposure.

During the transient events at the beginning ("second contact") and end ("third contact") of totality, a fixed aperture and shutter speed setting are appropriate, with a rather fast frame-to-frame timing. During totality, the film load of the camera should be spaced out over the balance of the 36 exposure magazine.

So, what we want the camera to do with its "n" exposures during the eclipse is to use a programmed sequence. The diagram of figure 1 shows a sequence that might have been ideal for me during the 1979 eclipse's 138 seconds of totality. In this figure, the events start at the last sliver of crescent sun when the filter is removed and a manual input starts the hypothetical computer sequence. Six shots are budgeted at 1/2 second intervals for the initial transient phenomenon called "Baily's beads" or "the diamond ring," depending upon the details of the sun shining through the lunar mountains.

The ideal case would then expose 24 frames at a uniform rate, covering an up and down sequence of exposure speeds. Finally, as the first bit of the departing transient starts to happen, the remaining six exposures would be used to capture the third contact "Baily's beads" or "diamond ring" effects as they occur. This would completely fill a single 36 exposure magazine of Kodak's excellent ASA 400 Ektachrome slide film. It sounds like a job for a microcomputer system as timing and control element, with suitable photographic peripherals.

How would this programmed sequence be possible? We want to use as much standard equipment as possible, for the purpose of reliability and to avoid total reinvention of the wheel. Fortunately, in contemporary photography, the motor drive is becoming an inexpensive and common accessory for the 35 mm SLR (single lens reflex) camera. This solves the problem of moving the film between frames. We need only set the motor drive on automatic and then the camera will take a picture and move the film to the next frame every time the exposure button is pushed. We must merely get the computer to push the shutter release button according to the timing diagram.

A relatively simple adaptation of a cable release to a solenoid actuator will serve to link the exposure button to the microcomputer sequencer. A suitable solid state relay power driver output from the computer will then press the button to take each picture. This, however, does not solve the problem of adjusting the shutter speed. Based on the current marketing literature of Nikon, I can get automated aperture control from an external source, but not control of the exposure time. Thus, the adaptation of my F2A camera will require careful thought and craftsmanship, of the same sort required for any other 35 mm camera body.

We will need a more elaborate combination of mechanical and electronic skills for this part of the operation. The adaptation of the camera shutter
I've finally found a personal computer I respect. It's not surprising that professionals get excited about the Compucolor II. It's a totally-integrated 8080A system with full color graphics display, built-in 51K mini-disk drive, and the best cost performance ratio available in a personal computer.

The complete system is only $1495. And that price includes 8K user RAM, RS-232C compatibility and random access file capabilities.

Our 8 foreground and background colors will boost your comprehension, while introducing you to an exciting new dimension in BASIC programming. The vector graphics have 16,484 individually-accessible plot blocks. And the 13" diagonal measure screen gives you 32 lines of 64 ASCII characters. You also have the flexibility that comes with 16K Extended Disk BASIC ROM.

Compucolor II offers a number of other options and accessories, like a second disk drive and expanded keyboard, as well as expandability to 32K of user RAM. Of course we also have a whole library of low-cost Sof-Disk™ programs, including an assembler and text editor.

Visit your nearest computer store for details. And while you're there, do some comparison testing. With all due respect to the others, once you see it, you'll be sold on the Compucolor II.

Compucolor Corporation  Post Office Box 569  Norcross, Georgia 30071  Telephone 404/449-9996

Unretouched photo of screen

* U.S. Domestic price
speed control to computer control requires machining skills with an amateur's lathe and milling setup.

The adapter is based upon a metal bracket which screws onto the camera at the tripod mounting socket which is standard on all 35 mm cameras. This bracket is set up with suitable spacers so that it will mount to the camera in a reliably repeatable fashion. The bracket can be mounted in the tripod socket since the telephoto lens or telescope used during an eclipse has its own mount on either a tripod or a true equatorial telescope base.

The adapter plate is then used to mount the small DC instrumentation motor and gearbox, which creates a reasonably high torque from the light duty motor. With my Nikon F2A, I need to turn the shutter speed control through an angle of approximately 10 degrees in about 1/4 to 1/2 of a second in order to accommodate the timing diagram of figure 1.

The output of the gearbox is a shaft which lines up with the axis of rotation of the shutter speed control. To this shaft is attached an optical position sensor created by passing the edge of a thin brass disk through an optoelectronic interrupter arrangement of the sort one can purchase from any optoelectronic company catalog. In the final stage of the custom fitting of this mechanism to the camera, each shutter position is marked on the disk and a small hole is punched in the disk. Thus, while the motor is turning the computer can tell when a given position has been reached, and the motor can be turned off.

The DC motor itself is controlled by a bidirectional electronic interface similar to the one shown in the article on the Terrapin Turtle by James A. Guptaon Jr. (“Talk to a Turtle,” June 1979 BYTE, page 74). This bidirectional interface allows us to turn the shutter speed knob to any setting, with the sensing of the shutter position returned by the optical interrupter. Use of a second interrupter for encoding of the first and last shutter speed settings will guarantee proper initialization and referencing of the speeds. This provides direct feedback of the limit stops in addition to the intermediate position information.

The one critical, unsolved problem in projecting this setup for my camera is the detail of driving the shutter speed control from the output of the gearbox. I will probably have to consider some potentially disastrous modifications to the camera. One possible method could be a tight fitting, carefully milled cylinder with ridges on its inner surface that would mate with the steps shown. During an eclipse, aperture setting of the camera cannot be controlled if long focus telephoto or astronomical telescope lens equipment is used.

![Figure 1: A timing diagram of an “ideal” 36 exposure sequence for a 138 second eclipse event, allowing two seconds before and after totality for transient phenomena. The horizontal axis of this figure is time in seconds, and the vertical axis is shutter speed of the camera, a discrete phenomenon with the steps shown. During an eclipse, aperture setting of the camera cannot be controlled if long focus telephoto or astronomical telescope lens equipment is used.](image-url)
The computer system which drives the camera during an eclipse is quite simple. The computer itself should be a dedicated 8 bit device with a suitable high level language program loaded into its local read only memory. Several 2708 or 2716 read only memory parts should suffice to store the systems software and the application program needed to control the telescope camera during a specific eclipse and to space out the 24 intermediate pictures during the estimated length of totality. Power requirements can be adequately handled by a single 6 or 12 V battery which also supplies power to the shutter speed control motor and the shutter tripping solenoid. The camera motor drive has its own dedicated NiCad battery pack which is totally independent of the computer. In a field situation it is assumed that batteries can be recharged through 110 V AC mains on the cruise ship or in land based hotels. If North American standard voltages are not available, the chargers can certainly be run through one of a number of standard converters available for world travelers.

Photo 4: One of the most unusual occurrences was the appearance of several hot air balloons in the sky at the time of the eclipse. While the balloonists must certainly have been having fun, some of the people watching from the ground were, no doubt, perturbed. Here is a kind of man made "cloud" phenomenon totally unpredictable by any meteorologist.

Figure 2: A conceptual sketch of possible homebrew machinery adapted to a 35 mm camera with motor drive, to allow automation of eclipse photography according to the timing diagram in figure 7. This fantasy was created by artist Ken Lodding.
Will I ever build this? At this time I can't predict if and when I will get around to building this sort of system. If I do, readers can be certain that there will be photographic documentation of the system. My immediate deadline might be to get the system working for the 1980 eclipse which occurs over the equatorial Atlantic Ocean, Africa, Indian Ocean, India and China on February 16. However, as this is written, I don't even know if I will go to see that event.

This camera automation computer is one of those applications of a small computer which is most appropriate. It has elements of the mechanical interfaces to electronics which are a necessary part of any practical robotic system, as well as elements of real time control akin to those needed for other practical uses of the small computer in home, laboratory and industry. It is the kind of system many of our readers are conceiving and building, whether it be for fun or for professional purposes. As time goes on, we can expect to see this kind of application documented in the form of articles with much greater detail than this editorial sketch. Conceiving, and then building this kind of application is when the fun of contemporary small computing reaches its highest level.

The More Things Change, The More They Stay The Same. . .

On April 20, 1979, BYTE Publications Inc and onComputing Inc became a part of McGraw-Hill Publications Co. Thus, as we neared completion of our fourth year as an enterprise, BYTE, and onComputing magazines joined Electronics, Aviation Week and Space Technology, and Data Communications to become key parts in a group of high technology magazines published by McGraw-Hill.

BYTE will continue to be published from offices in Peterborough NH, with the same staff and the same dedication to quality. Aside from such detail changes as the notation "A McGraw-Hill Publication" on our cover, readers can expect the editorial and advertising content of BYTE to continue under the same philosophy which has established our reputation in the past. Indeed, a major factor in our decision to affiliate with McGraw-Hill is their commitment to the independence of individual magazines.

An interesting statistic is that at this exciting time, BYTE's paid circulation of about 156,000 readers (May 1979 issue) makes it second only to Business Week in paid circulation among the more than thirty magazines published by McGraw-Hill.

We look forward at this point to a long and flourishing relationship with the people who form the McGraw-Hill enterprise.

. . . Carl Helmers

A Note About the Cover. . .

One of the interesting social phenomena of this eclipse was the appearance of a number of artificial clouds over the observation site: five or six different hot air balloons appeared over our site just at the time of totality. Photo 3 shows a wide angle shot that was intentionally overexposed during the partial phase of the eclipse just prior to totality. One of these artificial clouds is a dark object below and to the left of the sun in this picture. The telephoto shot shown in photo 4 captured one of these balloons in the sky to the west of the expedition site as they were drifting towards us, about 10 minutes before totality.

At the time of totality, I distinctly remember looking up and seeing two bright objects in the sky. One object was the eclipsed sun, and the second object, at about the same position as the dark balloon in photo 3, was one of the balloons with its propane flame shining a brilliant orange color. A man-made fire was complementing the eclipsed embers of the sun.

Combining the eclipse automation theme of this month's editorial with the hot air balloons actually observed, and the weather analysis and measurement themes of two of this month's articles, artist Robert Tinney has created a fantasy on eclipses, hot air balloons and weather for this month's cover. The dramatic effect of cumulonimbus thunderstorm clouds was used in place of the rather dull, high, thin cirrus cloud layer which partially obscured the 1979 eclipse as viewed from central Montana. And perhaps the hot air balloonists should have their heads examined for departing into this imagined thunderstorm, inexorable timing of an eclipse or not. But the resulting oil painting is an incomparable work of beauty, celebrating an uncommon event unique to our spaceship earth, its sister planet the Moon and a technological civilization.
Now you can put your S-100 system solidly into a full-size, single/double density, 600K bytes/side disk memory for just $1149 complete.

DISCUS/2D™ single/double density disk memory from Thinker Toys™ is fully equipped, fully assembled, and fully guaranteed to perform perfectly.

DISCUS/2D™ is a second generation disk memory system that's compatible with the new IBM System 34 format. The disk drive is a full-size Shugart 800R, the standard of reliability and performance in disk drives. It's delivered in a handsome cabinet with built-in power supply.

The S-100 controller utilizes the amazing Western Digital 1791 dual-density controller chip... plus power-on jump circuitry, 1K of RAM, 1K of ROM with built-in monitor, and a hardware UART to make I/O interfacing a snap.

The DISCUS/2D™ system is fully integrated with innovations by designer/inventor George Morrow. Software includes BASIC-V™ virtual disk BASIC, DOS, and DISK-ATE™ assembler/editor. Patches for CP/M* are also included. CP/M; MicroSoft Disk BASIC and FORTRAN are also available at extra cost.

DISCUS/2D™ is the really solid single/double density disk system you've been waiting for. We can deliver it now for just $1149. And for just $795 apiece, you can add up to 3 additional Shugart drives to your system. Both the hardware and software are ready when you are.

Ask your local computer store to order the DISCUS/2D™ for you. Or, if unavailable locally, write Thinker Toys,™ 5221 Central Ave., Richmond, CA 94804. Or call (415) 524-2101 weekdays, 10-5 Pacific Time. (FOB Berkeley. Cal. res. add tax.)

*CP/M is a trademark of Digital Research.

Morrow makes disk memory for

Thinker Toys™
Graphic Input of Weather Data

Stephen P Smith
106 E Clearview Av
State College PA 16801

Photo 1: Infrared and visible light photographs like this one are the primary tool of the satellite meteorologist. Cloud formations help locate rain. Temperature data from the infrared images indicates intensity. Photograph courtesy NOAA National Weather Service.
The fact that everyone complains about the weather, but nobody does anything about it, is well-known. Weather forecasting is still more art than science.

Manual techniques still yield the best quantitative weather predictions when compared with the largest computer systems processing a wealth of satellite, radar, and ground station data. However even modest data processing equipment can be an important tool for the meteorologist. I'll show how the combination of a small personal computer and a Summagraphics Bit Pad graphics tablet simplifies the processing of rainfall estimates for a regional data base. The application is an interesting one, and the BASIC language software developed will be useful in any system employing a Bit Pad for data entry.

Locally, rainfall can be measured with simple gauges. The heavy showers common during the prime growing and flood seasons have irregular distributions, so local measurements may be inadequate for regional use. Agricultural planners need to know how much rain has fallen over a specific growing area. Hydrologists working on flood warning and control need to know how much has fallen within a given watershed. Both groups need this information broken down into relatively small elements of time and area, perhaps for each 24 hour period and for each 10 kilometer square. To achieve this detail, tools in addition to rain gauges must be used.

The first of these tools is ground based radar. Most of us have seen weather radars operating on television news broadcasts. Rainfall reflects the radar signal and provides a visual display for the operator, similar to figure 1. Showers can be located accurately, and relative intensity can be determined. Unfortunately, even highly calibrated radars have difficulty measuring actual amounts of rain, and most weather radars are not well calibrated for this application. Radar coverage is also not complete over all areas of the country.

A second tool, satellite imagery, has extended that coverage significantly (see figure 2). Geostationary satellites, which remain fixed over one point on the earth, provide pictures every half hour. Polar orbiting satellites, flying much closer to the earth, provide more detailed images several times a day. A trained meteorologist can identify cloud formations in pictures like photo 1. Several investigators primarily at the National Oceanic and Atmospheric Administration (NOAA, pronounced like Noah) have developed schemes to estimate the rainfall beneath these clouds. (see references 1 and 2).

Text continued on page 20
We've worked like mad to
Now, we're cooking. Our boys in the lab have turned circuit chefs these past three months to create a smorgasbord of deliciously assembled boards to support your APPLE II*, TRS-80* or S-100 bus systems. Feast your eyes on our monstrous selection in the menu below. Then, order enough to satisfy your hunger for experimentation for months to come. Contact your local computer store, or call us direct. Bon appetit!

California Computer Systems
309 Laurelwood Road · Santa Clara, CA
(408) 988-1620 · 95050

So Nobody Goes Away Mad.

<table>
<thead>
<tr>
<th>APPLE II Delicacies</th>
<th>APPLE III Buffets</th>
</tr>
</thead>
<tbody>
<tr>
<td>7114: APPLE ROM...</td>
<td>7470: APPLE 3½ Digit BCD Analog-to-Digital Converter...</td>
</tr>
<tr>
<td>7120: APPLE Synchronous Serial Interface...</td>
<td>7510: APPLE Extender...</td>
</tr>
<tr>
<td>7710: APPLE Asynchronous Serial Interface...</td>
<td>7550: APPLE Wire Wrap Board...</td>
</tr>
<tr>
<td>7720: APPLE Parallel Interface...</td>
<td>7590: APPLE Etch Board...</td>
</tr>
<tr>
<td>7811: APPLE Arithmetic Processor...</td>
<td>7510: APPLE Solder Tail Board...</td>
</tr>
<tr>
<td>7490: APPLE GPIB IEEE 488 Interface...</td>
<td>7640: APPLE TRS-80 16K Add-On Memory...</td>
</tr>
</tbody>
</table>

# S-100 Bus Fare

| 2710 MXVI 16K Static RAM Board... | 2500 Wire Wrap Board... |
| 2730 4P10 Four Parallel I/O Board... | 2590 Etch Board... |
| 2520 Extender Terminator... | 2590 Solder Tail Board... |
| 2540 Wire Wrap Box... | 2590 All-Metal Mainframe Box... |

*APPLE II is a registered trademark of APPLE Computers, Inc.
*TRS-80 is a registered trademark of Radio Shack, a Tandy Co.
Figure 3: The result of the meteorologist's art is a map like this one. The curves are called isohyets. They define bands of equal rainfall in the same way that a contour map shows the height of the land.

Text continued from page 17:

The job of tying together the satellite information with ground station and radar data quickly falls to computers. The manual work should be made as simple as possible, with data provided to the computer in a format designed for the convenience of the meteorologist. The computer should accept this data, digitize it, manipulate it, store it, and produce reports in a format designed for agricultural planners, hydrologists, and other users. Programs of this type have typically been run on medium sized mainframe computers. For data entry and conversion, and limited report generation, however, there are a lot of good microcomputer applications.

Let's examine a specific example. We will accept rainfall estimates and process them to produce reports of accumulation in each 10 km square over a region of about 150,000 km² (i.e. 30 by 50 squares). It would be nice if a digitized satellite photograph could be input to a computer to generate rainfall estimates directly, but that is well beyond the current state of the art. Attempts have been made to input manual interpretations of cloud formations seen in the photos and have the computer evaluate their rain potential (see reference 3). This, too, lacks the necessary precision. The best estimates come from the evaluations of an experienced meteorologist working with satellite photos, and using ground station reports and radar data as a supplement.

In our system, the rainfall estimates are submitted as maps on which bands of equal rainfall will be drawn. Figure 3 shows a sample. The bands are called isohyets and are similar to the isobars, or lines of equal barometric pressure, also used by weather people.

A more familiar analogy might be a contour map. If locations on the ground are identified by XY coordinates, the rainfall rate can be thought of as a Z coordinate analogous to height at that point. Converting the isohyets to XYZ coordinates compatible with a grid of ten kilometer squares is the prime function of our processing system.

For a few locations, this could be done manually. The XY point could be located on the map and entered at a terminal along with the value of the isohyet in which it fell. When the number of points runs into the hundreds, however, and the data must be entered at half hour intervals, the manual approach becomes unworkable. A technique for rapidly entering the isohyets, automatically converting to XYZ format, and summing the entries over time must be devised.

Isohyets can be entered directly from the maps using a device called a graphic tablet. The map is placed on the tablet and a stylus is used to trace the outline of each isohyet. The tablet senses the position of the stylus, and the signals are electronically transmitted to a controller and converted to XY data. This type of equipment can resolve positions
North Star Announces —
Double Density x 2 Sides = Quad Capacity!
The North Star Horizon now delivers quad capacity by using two-sided recording on our new mini drives! That's 360,000 bytes per diskette! A four drive North Star system accesses over 1.4 megabytes of information on-line! Think of the application flexibility that so much information storage can give you!
North Star has quadrupled the disk capacity of the Horizon computer but prices have increased a modest 15 percent. On a dollar per byte basis, that's a bargain that is hard to beat!
The proven North Star disk controller was originally designed to accommodate the two-sided drives. North Star DOS and BASIC are upgraded to handle the new capacity, yet still run existing programs with little or no change. Of course, single sided diskettes are compatible with the new disk system.

North Star Horizon Computer Prices (with 32K RAM):
- Horizon-1-32K-Q $2349
- Horizon-2-32K-Q $2999
- Horizon-1-32K-D $2099
- Horizon-2-32K-D $2549

Get both sides now! Quad capacity is available from your North Star dealer.
If you want your machine to play, you get quality mini magnetic media. And at Verbatim Corporation, the whole message in quality. Would Victor Ruge like you? For the name of your nearest Verbatim distributor, call toll free: (1) 800-526-6797. In California call: (408) 777-7771. Call collect.

Verbatim Corporation
1000 E. Fairview Avenue, San Jose, CA 95110
Phone: (408) 275-4400
TWX: 133-625-9320

SHUGART SERENADE

Specify Verbatim

If you want your machine to play, you get quality mini magnetic media. And at Verbatim Corporation, the whole message in quality. Would Victor Ruge like you? For the name of your nearest Verbatim distributor, call toll free: (1) 800-526-6797. In California call: (408) 777-7771. Call collect.

Verbatim Corporation
1000 E. Fairview Avenue, San Jose, CA 95110
Phone: (408) 275-4400
TWX: 133-625-9320

Next Byte Mode Mode Rate Rate Rate Status

<table>
<thead>
<tr>
<th>Next</th>
<th>Byte Received</th>
<th>Mode 2</th>
<th>Mode 1</th>
<th>Rate 4</th>
<th>Rate 2</th>
<th>Rate 1</th>
<th>Status Valid</th>
</tr>
</thead>
</table>

Figure 4: The computer usually talks to the Bit Pad in parallel mode. One 8 bit word is used to control the Bit Pad. The two most significant bits are part of the communications handshake. The next five provide software control, duplicating switches on the controller. The last bit, status valid, should be set when transmitting a command.

to 0.1 millimeter and can enter data as fast as the operator can trace the lines. Until recently, graphics tablets were expensive, typically $5000. With the introduction of the Bit Pad from Summagraphics, however, the price is now within the range of the personal computer experimenter for scientific and business applications. As of this writing, a complete unit with a tablet, stylus and controller can be purchased for $555.

Using the Bit Pad for our rainfall application involves a little hardware and some software to interface the graphics tablet to my Ohio Scientific computer. The hardware interface has largely been taken care of by Summagraphics. The software problem is a matter of scaling and accounting for misalignment of the maps during the digitizing process. The same techniques will apply to any Bit Pad application on an 8 bit microcomputer. Although BASIC may be too slow for some applications, I'll use it to illustrate this application, so the concepts will be available to the widest variety of users.

Before dealing with code, let's look at the hardware interface. The Bit Pad uses 8 bit parallel input and output as its standard. Serial communication at TTL (transistor-transistor logic) or RS-232 voltages is available as an option. With the option, it may be possible to place the Bit Pad between your serial terminal and computer in much the same way that a SwTPC cassette interface is installed. I have a parallel port based on a 6520 PIA (peripheral interface adaptor) on my central processor board, so I elected to use the parallel format. The physical connection consists of a cable with a DB-25P plug on each end. One end plugs into the Bit Pad's controller. The other connects to the PIA lines brought out to my computer's back panel. The installation could hardly be simpler.

The 8 input and 8 output signals flow along the cable. The Bit Pad receives commands from the computer in the format shown in figure 4. Each XY point is trans-
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

<table>
<thead>
<tr>
<th>Model</th>
<th>Single Density</th>
<th>Double Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACS 8000-1</td>
<td>½ Mb</td>
<td>1 Mb</td>
</tr>
<tr>
<td>ACS 8000-2</td>
<td>1 Mb</td>
<td>2 Mb</td>
</tr>
<tr>
<td>ACS 8000-3</td>
<td>1 Mb</td>
<td>2 Mb</td>
</tr>
<tr>
<td>ACS 8000-4</td>
<td>2 Mb</td>
<td>4 Mb</td>
</tr>
</tbody>
</table>

Brackets show disk capacity per standard two drive system. All models come standard with 32 Kb RAM and two 8" disk drives as shown above. Expansion to 64 Kb is $363 per 16 Kb. FPP, DMA, software optional. Dealer/OEM discounts available. Delivery: 30 days ARO, all models.

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

Circle 6 on inquiry card.

2338A WALSH AVENUE • SANTA CLARA • CA 95050 • (408) 244-5766
Five bytes are used to transmit data to the computer. The first two bits of each are handshaking signals. The first byte contains status information. The bit labeled FO is set whenever the stylus is in contact with the tablet. F1, F2 and F3 are used only with an optional cursor. The remaining four bytes hold data. The second and third provide a 12 bit X coordinate and the fourth and fifth provide a 12 bit Y coordinate. Both are measured in absolute units (0.1mm or 0.005 inches) from the lower left corner of the tablet.

Figure 5: The Bit Pad communicates with the computer in parallel mode. Five bytes are used to transmit data to the computer. The first two bits of each are handshaking signals. The first byte contains status information. The bit labeled FO is set whenever the stylus is in contact with the tablet. F1, F2 and F3 are used only with an optional cursor. The remaining four bytes hold data. The second and third provide a 12 bit X coordinate and the fourth and fifth provide a 12 bit Y coordinate. Both are measured in absolute units (0.1mm or 0.005 inches) from the lower left corner of the tablet.

![Handshaking diagram](image-url)

Figure 6: A handshaking arrangement insures proper data transfer between the Bit Pad and your computer. The Bit Pad performs its part of the procedure automatically. A simple BASIC or machine language routine will handle the computer's end.
MEET MINIMAX.™
THE MOST ADVANCED, LEAST EXPENSIVE COMPUTER EVER CREATED.

Minimax has the largest memory (108k internal, 2.4 megabytes external) and the fastest data transfer rate (2 seconds per 30k characters) of any microcomputer in the world.

Minimax also has the most advanced full screen data entry and editing capabilities plus the highest resolution graphics (240x512).

The Minimax CPU features 64 user definable opcodes and ROM resident software complete with Microsoft Basic, DOS operating system, FIFTH language interpreter plus a complete machine language monitor with Tiny Assembler, Disassembler, Dump Facility and Debugging Aid.

And Minimax has a software library unmatched in the industry.

If all this sounds unbelievable, just check out the price: $4495 for the 800 kilobyte Minimax I and $5995 for the 2.4 megabyte Minimax II.

Now that's unbelievable.
Write or call for complete ordering information today.
Dealer inquiries invited.

MINIMAX™
Compu/Think™
3260 Alpine Road, Menlo Park, CA 94025 (415) 854-2577

Circle 71 on inquiry card.
10 REM DATA ENTRY ROUTINE USING SUMMAGRAPHICS BIT PAD
20 REM WRITTEN IN MICROSOFT 6502 BASIC
30 REM STEPHEN P. SMITH, STATE COLLEGE, PA
40 DIM D(4),EW(20,20),NS(20,20),M(10)
50 P1=63844: REM ADDRESS OF PARALLEL PORT
60 REM
1000 REM GET A POINT FROM THE BIT PAD
1010 GOSUB 1100: REM FETCH A BYTE
1020 IF 01 < 128 GOTO 1010: REM IS IT THE FIRST OF 5
1030 FOR I=1 TO 4: REM IF YES, READ NEXT 4
1040 GOSUB 1100: REM FETCH A BYTE
1050 D(I)=D1-64: REM STRIP THE HANDSHAKE BIT
1060 NEXT I
1070 X=D(2)*64+D(1): REM FIND THE ABSOLUTE X POSITION
1080 Y=D(4)*64+D(3): REM FIND ABSOLUTE Y POSITION
1090 RETURN
1100 REM HANDSHAKE ROUTINE
1110 POKE P1,128: REM SET "NEXT BYTE"
1120 WAIT P1+2,64,0: REM LOOK FOR "BYTE AVAIL" SET
1130 01=PEEK(P1+2): REM READ DATA
1140 POKE P1,64: REM RESET "NEXT BYTE" SET "BYTE RCVD"
1150 WAIT P1+2,64,64: REM LOOK FOR "BYTE AVAIL" RESET
1160 RETURN
3000 REM SET UP AUTOSCALING
3010 GOSUB 1000: REM GET X,Y, POINT A
3020 AX;X: AY;Y
3030 GOSUB 1000: REM GET X,Y, POINT B
3040 BX;X: BY;Y
3050 REM
3060 DATA 0,0,30,50: REM A & B IN GRID COORDINATES
3070 READ AG,AH,BG,BH
3080 S1=SQR((BG-AG)^2+(BH-AH)^2): REM DISTANCE AB IN GRID SYSTEM
3090 S2=SQR((BX-AX)^2+(BY-AY)^2): REM DISTANCE AB IN BIT PAD SYSTEM
3100 S=S1/S2: REM SET SCALE FACTOR
3110 T1=ATN((BY-AY)/(BX-AX)): REM ANGLE IN BIT PAD SYSTEM
3120 T2=ATN((BH-AH)/(BG-AG)): REM ANGLE IN GRID SYSTEM
3130 DT=T2-T1: REM ROTATION ANGLE
3140 DX=AX-(AG*COS(DT)-AH*SIN(DT))/S: REM X TRANSLATION
3150 DY=AY·(AH*COS(DT)+AG*SIN(DT))/S: REM Y TRANSLATION
3160 RETURN
3190 REM
3200 REM CONVERT ABSOLUTE X,Y TO GRID COORDINATES
3210 REM J IS NUMBER OF ISOHYET
3220 BX=X-DX: REM TRANSLATE X
3230 BY=Y-DY: REM TRANSLATE Y
3240 REM
3250 X=X-DX: REM TRANSLATE X
3260 Y=Y-DY: REM TRANSLATE Y
3270 X1=X*COS(DT)-Y*SIN(DT): REM ROTATE X
3280 Y1=Y*COS(DT)+X*SIN(DT): REM ROTATE Y
3290 EW(J,K)=X1*S: REM SCALE X AS EAST-WEST COORDINATE
3300 NS(J,K)=Y1*S: REM SCALE Y AS NORTH-SOUTH COORDINATE
3310 RETURN
4000 REM MENU BOARD ROUTINE
4010 REM
4020 DATA 0,.01,.02,.03,.04,.05,.06,.07,.08,.09,.10
4030 FOR I=1 TO 10: REM INITIALIZE 10
4040 READ M(I): REM VALUES FOR MENU
4050 NEXT I
4060 REM
4070 IF X>2794/20 GOTO 4060: REM USE LEFTMOST 5% OF TABLET
4080 N=Y/2794/10: REM MENU HAD 10 ELEMENTS
4090 R=M(N): REM Y POSITION SELECTS ONE
4100 RETURN

Listing 1: BASIC program to use the Bit Pad for entering data. This program inputs a point using the discussed handshaking method, automatically scales and rotates the point, and then allows you to perform operations using that point.

The distance between them gives us the scale factor. Their relative orientation tells the system how the map is positioned on the tablet. Figure 7 and caption give a general presentation of coordinate transformations. The subroutine which sets up this transformation begins at line 3000 of listing 1. At line 3200 absolute X and Y values are converted into grid coordinates and stored in a pair of arrays.

Using BASIC, the data is entered by touching the stylus to a number of points around each isohyet. The smooth curves on the map are approximated in the computer by polygons as in figure 8. A larger number of points produces a better approxi-

26 July 1979 - BYTE Publications Inc
“Our inventory is our existence. Think we’d trust it to anything less than Scotch® Brand Diskettes?”


Scotch Diskettes are the diskettes you can depend upon with the information your business depends upon.

Each one is tested and certified error-free before it leaves our factory. Because we know nothing less than perfection is acceptable for your vital business data.

Scotch Diskettes are available in regular or mini sizes, compatible with almost any system.

To find out where you can find Scotch Diskettes, call toll free: 800-328-1300.
(In Minnesota, call collect: 612-736-9625.) Ask for the Data Recording Products Division.

If it's worth remembering, it's worth Scotch Data Recording Products.
Figure 7: In the general coordinate transform there are two problems. The first is translation. Each point must be moved by the distance between the origins of the two coordinate systems. Because the distance is the same for every point, you need only know the coordinates of one point in both systems to compute the translation vector $dX, dY$. The second problem is rotation. The line between A and B makes a different angle with the X axis than with the $X'$ axis. Each point must be rotated by the difference. To compute the transformation, each point is multiplied by a rotation matrix, then added to the translation vector. The equation looks like this:

\[
\begin{bmatrix}
  X \\
  Y
\end{bmatrix} = \begin{bmatrix}
  \cos d & -\sin d \\
  \sin d & \cos d
\end{bmatrix} \times \begin{bmatrix}
  X' \\
  Y'
\end{bmatrix} + \begin{bmatrix}
  dX \\
  dY
\end{bmatrix}
\]

Would You Like to Participate in a Weather Reporting System?

A system is being installed in the state of Virginia that typifies what can be done with microprocessors. At each of seven remote stations, sensors collect data on wind, rainfall, temperature, etc.; and store it in the memory of a dedicated microcomputer. About once a day the data is transmitted to a central minicomputer for processing and integration with other data sources. The microcomputers are nicknamed the Seven Dwarfs. The minicomputer, not surprisingly, is called Snow White. It should also be no surprise to those who work with small computers that the first station to be installed was called Grumpy.

Now suppose that instead of just seven stations, a large number of personal computer owners attached some simple sensors to their systems, and were linked in a personal computer network (see February 1978 BYTE). Such a network is being tried, also in Virginia. Sensors are manually read; touch tone pads are used for communication. The interest of involved citizens is producing a valuable new resource for meteorologists, hydrologists and agricultural planners; and an interesting new application for readers of BYTE.

Figure 8: After digitizing with the Bit Pad, simple closed curves are approximated by polygons. More corners in the polygon produce a better approximation. Unused areas of the tablet can be used for a menu board. Touching the stylus in the appropriate area enters the indicated value. A software handler is required.

When all the isohyets have been entered, the computer can begin to assign a rainfall
There's been a lot of talk lately about intelligent terminals with small systems capability. And, it's always the same. The systems which make the grade in performance usually flunk the test in price. At least that was the case until the SuperBrain graduated with the highest PPR (Price/Performance Ratio) in the history of the industry.

For less than $3,000*, SuperBrain users get exceptional performance for just a fraction of what they'd expect to pay. Standard features include: two dual-density mini-floppies with 320K bytes of disk storage, 64K of RAM to handle even the most sophisticated programs, a CP/M Disk Operating System with a high-powered text editor, assembler and debugger. And, with SuperBrain's S-100 bus adapter, you can even add a 10 megabyte disk!

More than an intelligent terminal, the SuperBrain outperforms many other systems costing three to five times as much. Endowed with a hefty amount of available software (BASIC, FORTRAN, COBOL), the SuperBrain is ready to take on your toughest assignment. You name it! General Ledger, Accounts Receivable, Payroll, Inventory or Word Processing... the SuperBrain handles all of them with ease.

Your operators will praise the SuperBrain's good looks. A full ASCII keyboard with a numeric keypad and function keys. A non-glare, dynamically focused, twelve inch screen. All in an attractive desktop unit weighing less than a standard office typewriter. Sophisticated users will acclaim SuperBrain's twin Z-80 processors which transfer data to the screen at 38 kilobaud! Interfacing a printer or modem is no problem using SuperBrain's RS-232C communications port. But best of all, you won't need a PhD in computer repair to maintain the SuperBrain. Its single board design makes servicing a snap!

So don't be fooled by all the freshman students in the small systems business. Insist on this year's honor graduate... the SuperBrain.

*Quantity one. Dealer inquiries invited.
**EPROMs out at the touch of a finger.**

After programming a 2708 or 2716 EPROM, you won’t need a screwdriver to pry it out of SSM’s new PBI board equipped with Textool sockets. Just flip the lever and lift it out. And on the same board, there are 4 sockets waiting for 2708 or 2716 EPROMs that can be independently addressed to any 4k or 8k boundary above 8000 hex. Two boards in one.

PBI has two separate programming circuits so 2708 or 2716 (5v) type of EPROMs can be programmed without modifying the board. Programming voltage is generated on-board by a DC-DC converter; no need for an external power supply. Programming sockets are Dip Switch addressable to any 4k boundary. And complete software is provided for programming and verifying EPROMs.

With our Magic Mapping™ feature, unused EPROM sockets don’t take memory space, so you are never committed to the full 4k or 8k of memory. The board can be configured for 0 to 4 wait states. Use fast or slow EPROMs. All lines are buffered.

The PBI kit is available at over 150 retail locations or directly from SSM for $145.00 (with Textool sockets) or $125.00 (without Textool sockets). All SSM kits are backed by a 90 day warranty. Assembled, one year warranty.

SSM manufactures a full line of S-100 boards, including CPU, Video, I/O, RAM, EPROM, Music, Prototyping, Terminator, Extender and Mother boards. For complete details just send for our new, free brochure.

**PB1 2708/2716 Programmer & 4k/8k EPROM Board**

---

rate to each square in the grid. It begins with the area of heaviest rain, and determines which grid squares are surrounded by that isohyet (e.g., does the center of a square fall within the approximating polygon?). The heaviest rate is assigned to each of these squares. Beyond this area are bands of successively lighter rain. The grid squares surrounded by these isohyets, which have not already been included in another isohyet, are assigned the corresponding lighter rates. The process continues until an area of zero rainfall is encountered. The inverse situation, an area of light rain surrounded by heavier precipitation, does not occur in the sudden, convective storms this program was designed to monitor.

When all the grid squares have been assigned, the computer then holds a record of the rainfall during a one half hour period. When the next map is processed, the storm will have moved, and the distribution of rain will be different. These half hour records can be totaled over periods of any length to provide accurate accumulation data. The meteorologist need only be concerned with instantaneous rates, however. The computer, with the aid of the Bit Pad, handles the motion of the storm and the subtle effects of its changing shape and intensity.

As you may well imagine, updating 1500 grid squares 48 times a day could easily overwhelm many small systems with data. Access to rapid, random storage such as floppy disks is mandatory. The work is still within the capabilities of personal sized computers, but it begins to involve specific operating systems and algorithms which would be useful to only a few readers.

The BASIC routines developed here, however, apply to any Bit Pad application. They demonstrate how easy it is to use the Summagraphics graphics tablet for data entry. In doing so, I hope this article has also shown that a microcomputer, when teamed with some novel peripherals, can be used to process data for a challenging meteorological application.

**REFERENCES**

Introducing The SwitchBoard™ I/O Interface, the most complete interface available for S-100 systems designed by George Morrow exclusively for Thinker Toys™.

The SwitchBoard™ interface provides 4 parallel ports and 2 RS232/TTY serial ports. Plus strobe and attention ports. Plus, on-board facilities for 4K of optional static RAM and 4K of user-supplied EPROM.

And every port is switch-programmable for total flexibility in interfacing complex peripherals such as 12-bit daisywheel printers.

Each parallel port can be switched for input or latched output. Both serial ports can be switched to any of 16 baud rates from 110 to 19,200. Each strobe and attention port flip-flop can be switched for pos or neg pulsing.

And yet, The SwitchBoard™ Interface won’t hang you up on price or delivery. In kit form, it’s just $199. $259 assembled. 2114 4K static RAM option (4 MHz Z-80 compatible). $70.

Ask your local computer shop to place your order immediately for priority shipping. Or, if unavailable locally, order direct from Thinker Toys™, 5221 Central Ave., Richmond, CA 94804.

Or call for The SwitchBoard™ at (415) 524-2101 weekdays, 10-5 Pacific Time.

*Sockets provided, chip set optional.

A product of Morrow’s Microstuff for Thinker Toys™
CAST A SPELL.
WIN A SORCERER.
If you've written software in Altair Basic, you've written "spells" for the Exidy Sorcerer.
Now, make it pay off!

There's never enough software. Particularly good software. That's why Exidy is sponsoring a software contest where nobody loses.

Altair programs run on Sorcerer. The Sorcerer computer's Standard Basic is compatible with Altair 4K and 8K Basic. So our contest is open to programs—we like to think of them as "spells" or "Sorcery"—written in all three of those Basic versions.

Trade one of yours for one of ours. Just for entering a program we'll send you a new, professionally written and documented program. Free. It's a classic game of concentration that's a fun mind-stretcher for both kids and adults. Plus you'll get our new 20" by 24" color poster.

And maybe 99 more good programs. We'll publish a bound book of the best programs entered—up to 100 of them, with full credit to each author. If you enter you can have a copy for just the printing and mailing cost. And if your program is included, you get the book free.

WIN THIS EXIDY SORCERER.

And maybe a free Exidy Sorcerer: Submit one of the four programs judged "best," and win a free Sorcerer computer. (Or choose Sorcerer accessories of equal value.) There'll be one winner in each of the following categories: Business, Education, Fun & Games, and Home/Personal management.

Test-run your entry free. Take your program to any participating Sorcerer dealer if you want to give it a test run. At the same time, maybe you'll want to jazz up your program to take advantage of Sorcerer's state-of-the-art features. These include 512 by 240 high-resolution graphics; user-defined characters; and dual cassette I/O, among others.

You can turn in your entry right at the dealer's. And collect your poster and new program on the spot.

Enter now. Send us your entry with the coupon. Or visit your dealer. But cast your best spell at Exidy now. And see if you can't make a free computer appear on your doorstep.

RULES:
1) Entries, including documentation, must be printed by computer or typed double spaced on 8½ by 11 paper, with your name on every page.
2) Enter as many times as you like. This coupon, or a copy of it, must be completed and attached to all entries.
3) Enter at any participating Exidy Sorcerer dealer, or mail entries postpaid to the address on this coupon.
4) Entries must be received by midnight, Aug. 31, 1979. Winners will be notified by Nov. 30, 1979. For a list of winners, send a self-addressed, stamped envelope marked "Winners List" to the coupon address.
5) You warrant, by your signature on this coupon, that all program and documentation material included in your entry is entirely your own original creation, and that no rights to it have been given or sold to any other party, and you agree to allow Exidy to use, publish, distribute, modify, and edit it as it sees fit.
6) All entries become the property of Exidy, Inc. No entries will be returned, nor any questions answered regarding individual entries. No royalties, payments or consideration beyond the items set forth in this advertisement will be given to any entrant.
7) Judging will be by a panel of experts chosen by, and including representatives of, Exidy, Inc. Judges may assign programs to whichever entry category they consider appropriate. Decision of the Judges is final.
8) Employees of Exidy, Inc., its dealers, distributors, advertising agencies and media not eligible. Void where prohibited, taxed or restricted by law.

CITY STATE____ ZIP__
DAYTIME PHONE _____________ 
TITLE OF PROGRAM ____ _________ 
SIGNATURE----------- DATE ____

EXIDY, INC.
969 W. Maude Ave.
Sunnyvale, CA 94086

Gentlemen:
Here's my "spell." Send me my free program and poster. If I win, send my Exidy Sorcerer computer to:

NAME
ADDRESS
CITY____ STATE____ ZIP__
DAYTIME PHONE
TITLE OF PROGRAM
CATEGORY ☐ Business ☐ Fun & Games
☐ Education ☐ Home/Personal Management

SIGNATURE
DATE

Copyright 1979, Exidy, Inc.
The output of the voltage controlled oscillator and the super low frequency oscillator is a square wave which is supplied to the mixer and through the envelope selection logic to the envelope generator and modulator.

**Noise Generator and Filter**

Since so many sounds incorporate noise as an integral component, the 76477 includes a noise generator which can be set to produce pink or white noise by selection of the proper components. (Pink noise has a spectral intensity inversely proportional to frequency over a specified range. White noise is random and has constant energy for a unit bandwidth.) Further refinement of the desired noise range is accommodated through an external clock input applied to pin 3. Figure 3 illustrates this hookup.

The noise generator output is sent to the mixer.

**The Mixer and Envelope Selection**

Figure 4 shows how the mixer section of a sound generator works and specifically details the logic codes for the SN76477. The mixer is essentially a gating network which digitally combines the outputs from the super low frequency oscillator, voltage

---

**Figure 3:** The noise generator and filter section is composed of an external clock, two resistors, and a capacitor. The nominal value of $R_N$ is 47 k ohms and the minimum value for $R_{NF}$ is 7.5 k ohms.

**Figure 2:** The SLF (super low frequency) and VCO (voltage controlled oscillator) sections of the Texas Instruments SN-76477 complex sound generator. The desired frequency is selected by adjusting the resistor and capacitor circuits. The frequency is determined by the following formulas: super low frequency = $0.64/R_{SLF} \times C_{SLF}$ and voltage controlled oscillator = $0.64/R_{VCO} \times C_{VCO}$.

**Oscillators**

Figure 2 illustrates the two oscillator sections and equations for frequency selection. Figure 2a is an SLF (super low frequency generator) with a normal range of 0.1 Hz to 30 Hz. This super low frequency output is most often used to provide the input to the voltage controlled oscillator which runs at a higher frequency. Such a combination results in frequency modulated sound synthesis. A familiar example is a siren.

The voltage controlled oscillator can be externally controlled by grounding pin 22. The frequency is then governed by a 0 to 2.35 V signal applied to pin 16. Signals above 2.35 V will saturate oscillator output. As a further enhancement, the voltage controlled oscillator allows pitch control through a similarly ranged signal applied to pin 19.

---

**Figure 4:** Outputs of the super low frequency oscillator, voltage controlled oscillator, and noise generator are digitally selected. The control table indicates the output produced for any particular input.
Turning computer programmers into computer composers!

For the first time: Hard-to-obtain computer music material has been collected into one convenient, easy-to-read book.

The BYTE Book of Computer Music combines the best from past issues of BYTE magazine with exciting new material of vital interest to computer experimenters. The articles range from flights of fancy about the reproductive systems of pianos to Fast Fourier transform programs written in BASIC and 6800 machine language. Included in this fascinating book, edited by Christopher P. Morgan, are articles discussing four-part melodies, a practical music interface tutorial, electronic organ chips, and a remarkable program that creates random music based on land terrain maps!

ISBN 0-931718-11-2 $10.00

Buy this book at your favorite computer book store or order direct from BYTE BOOKS Add 60¢ per book for postage and handling

70 Main Street Peterborough, New Hampshire 03458
Figure S: By carefully choosing what signals are combined, a variety of different types of sounds can be produced. Figure Sa shows a combination of the super low frequency generator and the voltage controlled oscillator producing a sound such as a siren. Figure Sb combines the super low frequency oscillator and the noise generator to generate a sound such as a steam engine. In figure Sc, the voltage controlled oscillator and noise generator are mixed together to form a faster on and off pulsing than produced using the super low frequency generator. When the inhibit one shot is mixed with noise (figure Sd) the resulting sound would sound like a gun being fired.

The individual outputs of the voltage controlled oscillator, super low frequency oscillator and noise generator are selected with codes of 000, 001, and 010 respectively, as shown in the chart accompanying figure 4. The true value of this device is demonstrated when complex sounds are produced by combining these three sources and utilizing the inhibit for emphasis.

Figure 5a shows how the voltage controlled oscillator can be modulated by the super low frequency oscillator. As mentioned, an example of this is a siren. If, on the other hand, the super low frequency oscillator were programmed as in figure 5b, and mixed with the noise generator, the mixer output would sound like the steam engine we previously discussed. For faster on/off pulsing of the noise generator, the voltage controlled oscillator could be selected, and would appear as in figure 5c.

The inhibit line, rather than being an actual sound source, controls the duration of the other three sections. The internal one shot, triggering a 100 ms burst of noise to a loud amplifier, would sound like a gun shot. This is detailed in figure 5d.

The combined mixer output then goes to the envelope generator and modulator where the amplitude (volume) of the output signal is tailored through proper attack and decay timing so that it will synthesize actual sounds accurately. A piano is most easily characterized by its sharp attack and very long decay. Figure 6 outlines the component calculations for these timed functions.

Manual Sound Synthesizer

The SN76477 is essentially an independent sound generator. This means that with a few discrete components it can independently synthesize the sound of sirens, phasers, guns, etc. A computer is not required to program this device and, in fact, with the exception of the envelope, mixer and inhibit selection inputs, it is not directly controllable with a microprocessor. An example of a typical hardwired circuit using the SN76477 is shown in figure 7. This circuit simulates the sound of a steam engine and a whistle. The timing components were selected by using the equations outlined in figures 2 thru 6. This circuit produces two sounds by multiplexing the mixer between the voltage controlled oscillator frequency and the super low noise outputs. Normally, with the push button open the super low frequency oscillator pulses the noise generator on and off, producing a chug-chug sound. When the button is pushed, oscillator IC2 multiplexes the integrated circuit to the voltage

Text continued on page 42
This exceptional print quality for $560!
The new Comprint model 912 printer for computers and terminals:

- Fast—225 characters/second (170 lpm)
- 80 character lines on 8½" wide paper
- Quiet, non-impact operation
- 6 month warranty

Available now at computer stores and industrial distributors.
Figure 6: The envelope selection (table) is determined by envelope select 1 and envelope select 2 (pins 1 and 28) as shown in the table. The attack and decay timing is determined by $R_{OS}$, $C_{OS}$, $R_{D}$, $C$, and $R_{A}$.

The envelope selection (table) is determined by envelope select 1 and envelope select 2 (pins 1 and 28) as shown in the table. The attack and decay timing is determined by $R_{OS}$, $C_{OS}$, $R_{D}$, $C$, and $R_{A}$.

<table>
<thead>
<tr>
<th>Envelope Select</th>
<th>Function Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
</tr>
</tbody>
</table>

H = high level  
L = low level or open

Figure 7: The Texas Instruments SN76477 is often used in a hardwired, dedicated device. One such use is simulated steam engine and whistle sound as shown here.
Fast Becoming the Reference Standard

Find out for yourself why The Computer Cookbook has become the standard reference book of those who have to put together computer systems—and make them work.

Only The Cookbook gives all the ingredients for a successful system—hardware and software. We tell you how big a computer you really need to get the job done—not what size they'd like to sell you. We tell you what you can do yourself—and what you shouldn't do yourself.

Thinking of a system for a small business? We tell you what you need for your scale of operation. Trying to evaluate "canned" programs? We'll tell the potential trouble spots in Accounts Receivable, Inventory, and other "packages."

Ready to do it yourself? Our 'helpful hints' can save you hours of frustrating trial-and-error learning. We give you program outlines and step-by-step recipes in our plain-English "Cookbook" language—recipes you can then adapt to your machine's own dialect.

Just dreaming? We tell you how to interface a hard disk to your S-100 microcomputer. We list information services you can get if you have a telephone modem. And we describe speech synthesizers, home control equipment and even robots you can play with.

Best of all: The Cookbook doesn't go out of date. Because we update it every two months—you merely plug the pages into our loose-leaf binder. Why bother with reference books that are already out of date? The Cookbook is always ahead of the pack.

All orders shipped UPS unless US Mail requested. Add $1.50 for shipping. (Overseas add $5). California residents add .98 sales tax at $15, $1.56 at $24. Checks should be payable to "The Computer Cookbook." VISA and Mastercharge accepted with card numbers and 4-digit expiration date.

Name

Company/Street

City_ _ _____ _____ _ _ ____ State_________ _ ___ Zip

( ) Mastercharge: Interbank No: _ _ ___ _ ___ _ ___ Card No: _ _ ___ _ ___ 
( ) Visa _ _ ___ _ ___ _ _ _ _ BOTH: Expiration date: _ _ / _ _

The Computer Cookbook / P.O. Box 4084 / Berkeley, CA 94704

Act Now. The Cookbook will cost $24 after July 1, 1979

Circle 52 on inquiry card.
Table 1: Designation of I/O (input/output) port assignments and associated component choices in the interface for the Texas Instruments SN76477 sound generator.

<table>
<thead>
<tr>
<th>Port 8</th>
<th>Bit 0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Envelope Select 1</td>
<td>Inhibit</td>
<td>Envelope Select 2</td>
<td>Mixer C</td>
<td>5</td>
<td>VCO Select</td>
<td>External VCO Select</td>
<td></td>
</tr>
<tr>
<td>Noise Filter Capacitor</td>
<td>150 pF</td>
<td>0.001 µF</td>
<td>0.01 µF</td>
<td>1.0 µF</td>
<td>4.7 µF</td>
<td>0.1 µF</td>
<td>unused</td>
<td>0.1 µF</td>
</tr>
<tr>
<td>Attack/Decay Capacitor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLF Capacitor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VCO Resistor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VCO Capacitor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Values which are sent to the output ports connected to the SN76477 interface to produce the indicated sound effects.

<table>
<thead>
<tr>
<th>Sound Effect Desired</th>
<th>Hexadecimal Value Sent to Output Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>32</td>
</tr>
<tr>
<td>Phaser</td>
<td>B6</td>
</tr>
<tr>
<td>Siren</td>
<td>82</td>
</tr>
</tbody>
</table>

Photo 2: A typical video based space exploration game could be enhanced by sound effects.

Photo 3: A look at the prototype circuit of figure 7 attached to the back of an I/O (input/output) board.

Text continued from page 38:
controlled oscillator only position approximately half the time. The voltage controlled oscillator is programmed to produce a whistle. Sufficient power to drive a speaker is facilitated by a two transistor complementary amplifier attached to pins 12 and 13.

Build a Computer Programmable Sound Generator Interface

While the SN76477 is not directly controllable by a computer as it exists, an interface between it and a computer can be designed which will give it some semblance of programmability. Figure 8 illustrates such an interface. Sound generation is programmed through three output ports, two of which control CMOS analog switches. These switches allow a variety of resistor and capacitor combinations to be selected. Total control requires three output commands from a BASIC or machine language program, and it is very easy to switch from a siren to a phaser gun sound when implemented as game sound effects. Photo 2 shows the video display of a typical space war game. Consider the sophistication that sound effects would add.

In the prototype, shown in photo 3, ports 8, 9, and 10 were chosen to drive the interface. Port 8 handles mixer and envelope selection; port 9 controls selection of components for the attack, decay and noise sections; and port 10 controls the SLF and VCO programming. The values chosen are nominal and will not allow unlimited sound synthesis. Potentiometers are added to facilitate fine tuning.

A More Sophisticated Programmable Sound Generator

The SN76477 is attached to a microcomputer largely through brute force. A far more sophisticated device has been.

Text continued on page 45
Figure 8: The SN76477 complete sound generator can be controlled by a computer. All capacitors are 100 V ceramic, except electrolytics which are 16 V or greater. All resistors are \( \frac{1}{2} \text{W} \pm 5\% \).
Figure 9a: Functional block diagram of the General Instrument AY-3-8910 programmable sound generator. The device is made by General Instrument Corp, Microelectronics Division, 600 W John St, Hicksville NY 11802.

Figure 9b: Map of the control registers of the AY-3-8910.
recently introduced and it is designed specifically as a bus controlled device. This new device is the AY-3-8910 from General Instrument. It uses no external components and synthesizes sounds totally by digital means. A functional block diagram is shown in figure 9a.

You'll notice a similarity between this programmable sound generator and the Texas Instruments device in that they both contain the same elemental sound synthesis components such as noise and tone generators. The real difference is that the General Instrument programmable sound generator is programmed through 16 read/write control registers rather than resistors and capacitors. These registers appear as 16 sequential memory mapped I/O (input/output) locations to the controlling processor.

The AY-3-8910 incorporates a noise generator, three tone generators, three mixers, an envelope generator and three digital to analog converters for amplitude control. An added benefit is the inclusion of two decoded I/O ports which are available for other external applications. All subsystems are controlled through the control register array.

The device is specifically designed to interface with the General Instrument CP1600 series of microprocessors but it can be easily accommodated by others. Figure 10 illustrates this simple attachment. A bidirectional address/data bus, DA0 thru DA7, provides the necessary communication path. Since there are 16 registers, only four bits of address are actually used, and A8 and A9 serve more as device select lines by definition. BC1, BC2, and BDIR are the bus control lines and define bus direction, reading, and writing of register data. While an expensive circuit such as that shown in figure 11 can be used as the clock for both the processor and the programmable sound generator, they are basically independent and can be different rates. The programmable sound generator clock is primarily used for the sound synthesis. The reset line clears all registers.

For all practical purposes, signal line BC2 is unnecessary and can be tied to +5 V. The read/write control logic is shown in table 4.

The timing of BC1 and BDIR control lines are shown in figure 12. Data transfer is carried out by strobing these lines, while the

Table 3: Values which are loaded into the control registers of the General Instrument AY-3-8910 sound generator in order to produce the indicated sound effects.
Figure 10: Typical microprocessor to programmable sound generator interface.

Figure 11a: A simple clock generator which can be used as a clock for the processor and the programmable sound generator.

Figure 11b: A typical audio output interface for driving a speaker from the programmable sound generator.

<table>
<thead>
<tr>
<th>Number</th>
<th>Type</th>
<th>+5 V</th>
<th>GND</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC1</td>
<td>CD4069</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>IC2</td>
<td>4013</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>IC3</td>
<td>LM386</td>
<td>6</td>
<td>4</td>
</tr>
</tbody>
</table>
FUN & GAMES

PLUS
SERIOUS SYSTEM SOFTWARE

AT YOUR COMPUTER RETAILER NOW

TRS-80, Apple II, Sorcerer, SOL and Southwest Tech 6800 owners: get more out of your personal computer.

Get into action with G2 Bullseye!, Sea Battle, Confrontation and Attack!

Sharpen your analytical abilities with G2's Outwit I, Outwit II and Mind Bender.

Take a chance with G2's Beat the House. Check out your health with Clinic.

And enjoy the challenging experience of two new G2 computer simulations: The Market and Wildcatting.

Or get serious with three powerful new languages: Level III Basic for the TRS-80, Extended Basic for the SOL, and Standard Basic for the Southwest Tech 6800. All written by Microsoft—the Basic wizards. Exclusively from G2.

Our software has more so your computer does more. Great programming. Highest quality cassette. The most comprehensive instruction manual available. Plus source listing print-out of every application program in Basic. You can learn how the programs were written. We even encourage you to do your own re-programming to improve your skills!

G2 software is available from computer retailers nationwide. If your local retailer doesn’t have it, ask him to become a G2 dealer by calling toll free:

(800) 538-8540 (U.S.A.) or (800) 672-8691 (California).

THE REASON YOU BOUGHT YOUR COMPUTER.

A Product of G2 Corporation

BYTE July 1979
Table 4: Summary of the read/write control logic needed to control the AY-3-8910 sound generator.

<table>
<thead>
<tr>
<th>BDIR</th>
<th>BC2</th>
<th>BC1</th>
<th>Function</th>
<th>CP1600 Function Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Inactive</td>
<td>NACT</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Read from PSG</td>
<td>DTB</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Write to PSG</td>
<td>DWS</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Latch Address</td>
<td>INTAK</td>
</tr>
</tbody>
</table>

Table continued from page 45:

Address/data bus contains the pertinent contents. These pulses should be short and one processor clock cycle should suffice.

Tone Select

The registers are divided into six categories, and numbered in base eight:

- Tone generators: \( R_0 \) thru \( R_5 \)
- Noise generator: \( R_6 \)
- Mixer control: \( R_7 \)
- Amplitude control: \( R_{10} \) thru \( R_{12} \)
- Envelope control: \( R_{13} \) thru \( R_{15} \)
- I/O ports: \( R_{16} \) and \( R_{17} \).

Tones are square waves produced by dividing the input clock by 16, then counting that result down by a programmed 12 bit tone-period value. The 12 bit value, defined by the coarse and fine tune registers, is a combination of the two control registers. The 12 bits represent period and \( T = \frac{1}{\text{frequency}} \). The higher the register value, the lower the tone. Register contents range from 000000000001 (divide by 1) to 111111111111 (divide by 4095). With a 2 MHz clock the frequencies would be 125 kHz and 30.5 Hz respectively.

The other parameters, such as noise, mixers, amplitude, and envelope controls, are chosen in a similar manner. The actual programming technique is beyond the scope of this introduction to the AY-3-8910, and I suggest that interested readers send inquiries to General Instrument.

Connecting the AY-3-8910 to the S-100 Bus

Figure 13 shows how an AY-3-8910 programmable sound generator can be connected as an I/O device on the S-100 8080 compatible bus. Switches SW1 through SW6 define the starting I/O address of the 16 programmable sound generator registers.

<table>
<thead>
<tr>
<th>Number</th>
<th>Type</th>
<th>+5 V</th>
<th>GND</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC1</td>
<td>7485</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC2</td>
<td>7485</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC3</td>
<td>7404</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>IC4</td>
<td>7402</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>IC5</td>
<td>7400</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>IC6</td>
<td>7400</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>IC7</td>
<td>74148</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC8</td>
<td>74LS367</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC9</td>
<td>74LS367</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC10</td>
<td>74LS367</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC11</td>
<td>74LS367</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>IC12</td>
<td>AY-3-8910</td>
<td>40</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 12: Programmable sound generator bus timing logic.
Figure 13: The AV:38910 can be connected to the S100 bus. SW1 thru SW6 define the starting address of the 16 control registers. The power pin assignment is shown in the table at left.
LATCH ADDRESS ROUTINE
PORTADDR EQU 80H ; ADDRESS TRANSFER PORT ADDRESS
PORTDATA EQU 81H ; DATA TRANSFER PORT ADDRESS

; THIS ROUTINE WILL TRANSFER THE CONTENTS OF
; 8080 REGISTER C TO THE PSG ADDRESS REGISTER
PSGBAR MOV A,C ; GET C IN A FOR OUT
OUT PORTBAR ; SEND TO ADDRESS PORT
RET

WRITE DATA ROUTINE
; ROUTINE TO WRITE THE CONTENTS OF 8080 REGISTER B
; TO THE PSG REGISTER SPECIFIED BY 8080 REGISTER C

PSGWRITE CALL PSGBAR ; GET ADDRESS LATCHED
MOV A,B ; GET VALUE IN A FOR TRANSFER
OUT PORTDATA ; PUT TO PSG REGISTER
RET

READ DATA ROUTINE
; ROUTINE TO READ THE PSG REGISTER SPECIFIED
; BY THE 8080 REGISTER C AND RETURN THE DATA
; IN 8080 REGISTER B

PSGREAD CALL PSGBAR ; GET ADDRESS LATCHED
IN PORTDATA ; GET REGISTER DATA
MOV B,A GET IN TRANSFER REGISTER
RET

Listing 1: Routines written for the 8080 microprocessor to operate the AY-3-8910 programmable sound generator.

LATCH ADDRESS ROUTINE
; AT ENTRY, B HAS ADDRESS VALUE
LATCH CLRA
STAA 8005 ; GET D IR A
LDAA # FF
STAA 8004 ; OUTPUTS
LDAA # 4
STAA 8005 ; GET PERIPHERAL A
STAB 8004 ; FORM ADDR
STAA 8006
CLA
STAA 8006 ; LATCH ADDRESS
RTS ; RETURN

WRITE DATA ROUTINE
; AT ENTRY, B HAD DATA VALUE
WRITE CLRA
STAB 8004 ; FORM DATA
LDAA # 6 ; DWS
STAA 8006
CLA
STAA 8006 ; WRITE DATA
RTS ; RETURN

READ DATA ROUTINE
; AFTER READ, B HAS READ DATA
READ CLRA
STAA 8005 ; GET D IR
STA A 8004 ; INPUTS
LDAA # 4
STA A 8005 ; GET PERIPHERAL
DECA
STA A 8006 ; READ MODE
LOA B 8004 ; READ DATA
CLA
STA A 8006 ; REMOVE READ MODE
RTS ; RETURN

Listing 2: Routines coded for the 6800 microprocessor to operate the AY-3-8910.

Figure 14: Connecting the AY-3-8910 to a 6800 system through a 6820 programmable interface adapter is easier than interfacing the S-100 bus.
Reading and writing from it is as illustrated.

A less complicated hardware interface is attained by using a peripheral interface adapter such as the 6820. Figure 14 demonstrates a technique which can be used for 6800 systems. The considerable difference in hardware complexity should in no way imply lack of ability using the 8080. If the S-100 bus is ignored and a 8255 programmable peripheral interface is used instead, it would result in a circuit similar to figure 14.

In Conclusion

I have briefly presented two methods of sound synthesis. While both are simple to implement, it is easy to recognize that the Texas Instruments part is more applicable in dedicated designs while the General Instrument device is for general synthesizer applications. It is not inconceivable that the AY-3-8910 could produce almost any sound, and it is a natural for use with a music interpreter running on a microcomputer. Perhaps the next famous composer will not direct a 150 piece orchestra but, rather, a trio of microcomputers controlling a bank of AY-3-8910s.

Circuit diagrams and drawings pertaining to the AY-3-8910 were provided courtesy of General Instrument Corp.

ATTENTION CREATIVE ELECTRONIC ENGINEERS, MICROCOMPUTER PROGRAMMERS AND TECHNICIANS

TRAVEL TO THE NORTHEAST TO WESTERN NEW ENGLAND TO SCENIC EAST LONGMEADOW, MA. NESTLED IN THE FOOTHILLS OF THE BERKSHIRES AND A WHOLE NEW WORLD OF ELECTRONIC OPPORTUNITY AWAIT YOU.

The Milton Bradley Company has propelled itself into the electronic game and toy arena in a most dramatic and successful fashion in the last two years. This momentum is continuing in a most encouraging way. Consequently, we are seeking talented electronic personnel with both hardware and/or software experience and knowledge to expand our inhouse electronic product development capability. We have immediate openings for the following types of personnel:

**Senior Electronic Product Engineers**
Qualified candidates will have a minimum of ten years electronic consumer product experience with a BSEE degree or the equivalent thereof.

**Electronic Product Engineers**
This position requires a minimum of 5 years electronic consumer goods experience and a BSEE or equivalency.

**Software Engineers/Microcomputer Programmers**
3 to 5 years experience in microcomputer programming and knowledge of microcomputer assembly languages is required.

**Electronic Technicians**
1 to 3 years experience as a technician in an electronic consumer goods environment is preferred. An AS degree in electronics would be an asset.

In addition to scenic beauty in the immediate locale we offer proximity to skiing, sailing, surfing and several other quality factors of life.

The successful candidates will enjoy an attractive compensation package which includes a complete fringe benefit program.

Please send resume and salary requirements in confidence to:

**Director of Corporate Personnel Administration**
Milton Bradley Company
P.O. Box 3400 — Springfield, MA. 01101
Circle 233 on inquiry card.
An Equal Opportunity Employer M/F

**Come Help Us Celebrate the Child**

St. Jude Children’s Research Hospital continues its search for life-saving knowledge about childhood diseases. And this search continues because people care. Help us celebrate the child by sending your tax-deductible check or request for further information to St. Jude Children’s Research Hospital.

539 Lane Ave.,
Memphis, TN 38105.

**ST. JUDE CHILDREN’S RESEARCH HOSPITAL**

Danny Thomas, Founder

MILTON BRADLEY

An Equal Opportunity Employer M/F
Listing 1 is a short program that generates a fastmoving colorful display on your Apple II.

The program starts at hexadecimal location 0800 and resides in less than one page of memory. It may be entered in object form by use of the system monitor. To run the program, simply type: 800G. The display can be frozen by hitting any key on the keyboard. Hitting any key again will resume the action.

The program was written using Microproducts Editor/Assembler program for the Apple II.
some friendly advice; there simply is no other place to go.

We offer experience and
custom software, service, or just

Circle 66 on inquiry card. July 1979 © BYTE Publications Inc 53
A Model of the Brain for Robot Control

Part 2: A Neurological Model

In part 1 I described how sensory interactive, goal directed behavior can be generated and controlled by a multilevel hierarchy of computing modules. At each level of the hierarchy, input commands are decomposed into strings of output subcommands which form the input commands to the next lower level. Feedback from the external environment, or from internal sources, drives the decomposition process and steers the selection of subcommands so as to achieve successful performance of the task of reaching the goal. In this article I will address questions of what kind of neurological structures are believed to exist in the brain and what kind of computations, memory storage methods, and associative recall effects these structures seem to be performing.

Unfortunately, definitive experimental evidence about the structure and function of neurological circuitry in the brain is extremely difficult to obtain. Neurons, the brain's computing elements, are very tiny and delicate. It is hard to measure what is happening in them without damaging them or otherwise interfering with the flow of information related to their operation. Techniques do exist for measuring the activity of individual neurons and sometimes even observing the behavior of several neurons at the same time. There are also techniques which make it possible to monitor synchronized changes in the activity of large numbers of neurons.

However, the brain is such a complicated anatomical structure, with such a jumbled interconnection of different kinds of neurons being excited and inhibited by such a broad variety of chemical and electrical stimuli, that it is impossible to infer from these measurements any very sophisticated ideas about what mathematical functions are being computed or what procedures are being executed.

Neurons are as varied in size, shape, and type as trees and bushes in a tropical forest, and often are as closely intertwined and interconnected as a bramble patch overgrown with vines. Many of their most important information processing properties are statistical in nature, and these statistics may apply over ensembles of thousands of neurons.

The situation is further complicated by multiple feedback loops, some of which are confined to small, local clusters of neurons, and others which may thread through several entirely different regions of the brain. The result is that no one has yet been able to construct a clear picture of the overall information processing architecture in the brain. At present there exists no generally accepted theory which bridges the gap between hard neurophysiological measurements and psychological concepts such as perception and cognition.

Nevertheless, there is much that is known with certainty about the structure and function of at least some parts of the brain, particularly in the periphery of the sensory and motor systems. A great deal can be inferred from this knowledge. Furthermore, there is one area, the cerebellar cortex, where the geometry is sufficiently regular to enable researchers to positively identify a
The September '77 and March '79 covers of BYTE are now each available as a limited edition art print, personally signed and numbered by the artist, Robert Tinney.

These prints are strictly limited to a quantity of 750 for each cover, and no other editions, of any size, will ever be published. Each print is 18" x 22", printed on quality, coated stock, and signed and numbered in pencil at bottom.

The price of each print is $25. This includes 1) a signed and numbered print; 2) a Certificate of Authenticity, also signed personally by the artist and witnessed, attesting to the number of the edition (750), and the destruction of the printing plates; and 3) first class shipment in a heavy-duty mailing tube.

To order your limited edition art print, fill out and mail the order form below.

Send me ______ “Breaking the Sound Barrier” prints at $25 each, and ______ “Trap Door” prints at $25 each. I understand this price includes Certificate of Authenticity and first class shipment.

☐ I have enclosed check or money order to Robert Tinney Graphics.
☐ Charge this to my Master Charge or Visa

Card #__________ Expires:______

Ship my print(s) to:
Name__________________________
Address________________________
City_________________ State______ Zip_______

Send order to:
robert tinney graphics
P.O. Box 45047 • Baton Rouge, LA 70895
The principal input to the cerebellar cortex arrives via mossy fibers.

The cerebellum, which is attached to the midbrain portion of the upper spinal cord and nestles up under the visual cortex, as shown in figure 1, is intimately involved with control of rapid, precise, coordinated movements of limbs, hands, and eyes. Injury to the cerebellum results in motor deficiencies, such as overshoot in reaching for objects, lack of coordination, and the inability to execute delicate tasks or track precisely with the eyes.

During the 1960s, advances in the technology of single cell recordings and electron microscopy made possible an elegant series of experiments by Sir John Eccles and a number of others. These experiments identified the functional interconnections between the principal components in the cerebellar cortex. A brief outline of the structure and function of the cerebellar cortex is shown in figure 2.

The principal input to the cerebellar cortex arrives via mossy fibers (so named because they looked like moss to the early workers who first observed them through a microscope). Mossy fibers carry information from a number of different sources such as the vestibular system (balance), the reticular formation (alerting, the cerebral cortex (sensory-motor activity), as well as from sensor organs which measure such quantities as position of joints, tension in tendons, velocity of contraction of muscles, pressure on skin, etc. It is possible to categorize mossy fibers into at least two classes based on their point of origin: one, those carrying information which may include commands from higher levels in the motor system; and two, those carrying feedback information about the results of motor outputs. Once these two sets of fibers enter the cerebellum, however, they intermingle and become virtually indistinguishable.

The feedback mossy fibers tend to exhibit a systematic regularity in the mapping from point of origin of their information to their termination in the cerebellum. It is thus possible to sketch a map of the body on the surface of the cerebellum corresponding to the origins of feedback mossy fiber information, as shown in figure 3. This map is not sharply defined, however, and has considerable overlap between regions due in part to extensive intermingling and multiple overlapping of terminations of the mossy fibers in the cerebellar granule cell layer. Each mossy fiber branches many times and makes excitatory (+) contact with several hundred granule cells spaced over a region several millimeters in diameter.

Granule cells are the most numerous cells in the brain. It is estimated that there are about 3 $\times 10^{10}$ granule cells in the human cerebellum alone. There are 100 to 1000 times as many granule cells as mossy fibers. Each granule cell is contacted by 5 to 12 mossy fibers and gives off a single output axon which rises toward the surface of the cerebellum. When it nears the surface this axon splits into two parts which run about 1.5 mm in opposite directions along the folded ridges of the cerebellum, making contact with a number of different kinds of cells in passage. These axons from the granule cells thus run parallel to each other in a densely packed sheet (hence the name, parallel fibers).

One of the cell types contacted by parallel fibers are Golgi cells (named for their discoverer). These cells have a widely spread dendritic tree and are excited by parallel fibers over a region about 0.6 mm in diameter. Each Golgi cell puts out an axon which branches extensively, making inhibitory (−) contact with up to 100,000 granule cells in its immediate vicinity, including many of the same granule cells which excited it. The dendritic trees and axons of neighboring Golgi cells intermingle so as to blanket the entire granular layer with negative feedback. The general effect is that of an automatic gain control on the level of activity in the parallel fiber sheet.

It is thought that the Golgi cells operate such that only a small and controlled percentage (perhaps as little as 1 percent or
less) of the granule cells are allowed above threshold at any one time, regardless of the level of activity of the mossy fiber input. Any particular pattern of activity on the mossy fiber input will produce a few granule cells which are maximally excited, and a great many others which are less than maximally stimulated. The Golgi cells suppress the outputs of all but the few maximally stimulated granule cells. The result is that every input pattern (or vector) is transformed by the granule layer into a small, and relatively fixed percentage, or subset, of parallel fibers which are active.

These active parallel fibers not only contact Golgi cells, but make excitatory contact with Purkinje cells (named for their discoverer) and basket and stellate cells (named for their shapes) through weighted connections (synapses). Each Purkinje cell performs a summation over its inputs and produces an output which is the output of the cerebellar cortex. The basket and stellate cells are essentially inverters which provide the Purkinje with negative weights that are summed along with the positive weights from parallel fibers.

Figure 2. The principal cells and fiber systems of the cerebellar cortex. Command and feedback information arrives via mossy fibers, each of which make excitatory (+) contact with several hundred granule cells. Golgi cells sample the response of the granule cells via the parallel fibers and suppress by inhibitory (−) contacts all but the most highly excited granule cells. Purkinje cells are the output of the cerebellar cortex. They sum the excitatory (+) effect of parallel fibers through weighted connections. They also receive inhibitory (−) input from parallel fibers via basket cell inverters. The strengths of these weights determine the transfer function of the cerebellar cortex. Climbing fibers are believed to adjust the strength of these weights so as to train the cerebellum.

Figure 3. A map of the surface of the cerebellar cortex showing the point of origin of mossy fiber feedback and ultimate destination of Purkinje cell output.
CMAC is defined by a series of mappings:

\[ S \rightarrow M \rightarrow A \rightarrow p \]

where:

- \( S \) is an input vector;
- \( M \) is the set of mossy fibers used to encode \( S \);
- \( A \) is the set of granule cells contacted by \( M \);
- \( p \) is an output value.

The overall mapping:

\[ S \rightarrow p \]

has all of the properties of a function:

\[ p = h(S) \]

as described in part 1. A set of \( L \) CMACs operating on the same input produces a mapping:

\[ S \rightarrow P \]

which has the properties of the function:

\[ P = H(S). \]
Meet Super Grip II, the great new test clip from A P Products.

New narrow-nose design makes it easy to attach on high-density boards. And now you can test ICs with only .040" between opposing legs.

New "duck bill" contacts are flat, won't roll off IC leads.

Open-nose construction enables probe at IC leg.

Pin rows are offset for easy attachment of probes.

Contacts are gold-plated phosphor bronze. "Contact comb" construction separates contacts with precision. No shorts.

Heavy-duty, industrial-grade springs for firm contact pressure—and a good grip when pulling ICs. They'll keep their spring indefinitely. No intermittents.

Steel pivot pin. Engineering-grade thermoplastic body molded around contact pins. Made to last.

New button-head pins keep probes from sliding off. (Straight pin models for logical connections.)

The new A P Super Grip II is, without question, the best way there is to troubleshoot DIP ICs. You get positive contacts. No intermittents. No shorts. Ever. So it's endlessly useful to you—and it's built to stay useful indefinitely.

Try one. You'll find 8, 14, 16, 16 LSI, 18, 20, 22, 24, 28, 36 and 40-pin models at your nearby A P store. (Make sure it's your A P store.) Need the address? Call (toll-free) 800-321-9668. And ask for our complete A P catalog, The Faster and Easier Book.

Faster and Easier is what we're all about.
We may describe the information encoded by mossy fibers as a vector $S = C + F$ where:

$$C = (s_1, s_2, \ldots, s_i)$$

is a vector, or list, of command variables;

and

$$F = (s_{i+1}, \ldots, s_N)$$

is a vector, or list, of feedback variables.

$+$ is an operator denoting the combination of two vectors defined by two lists of variables into a single vector or list of variables.

That is:

$$S = C + F$$

means that $S = (s_1, s_2, \ldots, s_i, s_{i+1}, \ldots, s_N)$.

Some of the elements of the command vector $C$ may define symbolic motor commands such as $<\text{REACH}>$, $<\text{PULL BACK}>$, $<\text{PUSH}>$, etc. The remainder of the elements in $C$ define arguments, or modifiers, such as the velocity of motion desired, the force required, the position of the terminal point of a motion, etc. Elements of the feedback vector $F$ may represent physical parameters such as the position of a particular joint, the tension in a tendon, the velocity of contraction of a muscle, the pressure on a patch of skin, and so on.

Mapping $S \rightarrow M$

The vector components of $S$ must be transmitted from their various points of origin to their destination in the cerebellar granular layer. Distances may range from a few inches to over a foot. This presents a serious engineering problem because mossy fibers, like all nerve axons, are noisy, unreliable, and imprecise information channels with limited dynamic range. Pulse frequency and pulse phase modulation (which the brain uses for data transmission over long distances) are subject to quantization noise and are bandwidth limited. Nerve axons typically cannot transmit pulse rates above two or three hundred pulses per second. Nevertheless, high resolution high bandwidth data is required for precise control of skilled actions.

The brain solves this problem by encoding...
each of the high precision variables to be transmitted so that it can be carried on a
greater number of low precision channels. Many mossy fibers are assigned to each input
variable such that any one fiber conveys only a small portion of the information
content of a single variable.

The nature of this encoding is that any particular mossy fiber will be maximally
active over some limited range of the variable that it encodes, and less than maximally
active over the rest of its variable's range. For example, the output of the mossy fiber
labeled a in figure 6 is maximally active whenever the elbow joint is between $90^\circ$ and
$120^\circ$ and is less than maximally active for all other elbow positions. The mossy fiber
labeled b in figure 6 is maximally active whenever the elbow angle is greater than
$160^\circ$. Now if there exists a large number of mossy fibers whose responses have a single
maximum but which are maximally active over different intervals, it is then possible to
tell the position of the elbow quite precisely by knowing which mossy fibers are maxi-
mally active. For example, in figure 7 the fact that mossy fibers a, b, and c are maxi-
mally active indicates that the elbow joint is between $118^\circ$ and $120^\circ$.

The CMAC models this encoding scheme in the following way: define $m_i$ to be the
set of mossy fibers assigned to convey the value of the variable $s_i$; define $m_i^*$ to be the
mossy fibers in $m_i$ which are maximally stimulated by a particular value of $s_i$. If for
every value of $s_i$ over its range there exists a unique set $m_i^*$ consisting of the set of values produced by the K quantizing
functions. For example (in figure 8), the value $s_1 = 7$ maps into the set $m_1^* = \{B, H,
P, V\}$.

A similar mapping is also performed on $s_2$ by the set of quantizing functions:

$2C_1 = \{a, b, c, d, e\}$
$2C_2 = \{f, g, h, i, k\}$
$2C_3 = \{m, n, p, q, r\}$
$2C_4 = \{s, t, v, w, x\}$

For example, the value $s_2 = 10$ maps into the set $m_2^* = \{c, j, q, v\}$. Now, if the $s_1$
component in figure 8 corresponds to the position of the elbow joint, the mossy fiber
labeled B will be maximally active whenever

\begin{align*}
\text{Figure 6. Typical responses of mossy fibers to the sensory variable they encode.}
\end{align*}

\begin{align*}
\text{Figure 7: Three different mossy fibers encoding a single sensory variable (elbow position). All three fibers maximally active simultaneously indicates that the elbow lies between 118° and 120°.}
\end{align*}

\begin{align*}
1C_1 & \{A, B, C, D, E\} \\
1C_2 & \{F, G, H, J, K\} \\
1C_3 & \{M, N, P, Q, R\} \\
1C_4 & \{S, T, V, W, X\}.
\end{align*}

Each quantizing function is offset from the previous one by one resolution ele-
ment. For every possible value of $s_i$ there exists a unique set $m_i^*$ consisting of the set of values produced by the K quantizing
functions. For example (in figure 8), the value $s_1 = 7$ maps into the set $m_1^* = \{B, H,
P, V\}$.

A similar mapping is also performed on $s_2$ by the set of quantizing functions:

$2C_1 = \{a, b, c, d, e\}$
$2C_2 = \{f, g, h, i, k\}$
$2C_3 = \{m, n, p, q, r\}$
$2C_4 = \{s, t, v, w, x\}$

For example, the value $s_2 = 10$ maps into the set $m_2^* = \{c, j, q, v\}$. Now, if the $s_1$
component in figure 8 corresponds to the position of the elbow joint, the mossy fiber
labeled B will be maximally active whenever

\begin{align*}
\text{Figure 6. Typical responses of mossy fibers to the sensory variable they encode.}
\end{align*}

\begin{align*}
\text{Figure 7: Three different mossy fibers encoding a single sensory variable (elbow position). All three fibers maximally active simultaneously indicates that the elbow lies between 118° and 120°.}
\end{align*}

\begin{align*}
1C_1 & \{A, B, C, D, E\} \\
1C_2 & \{F, G, H, J, K\} \\
1C_3 & \{M, N, P, Q, R\} \\
1C_4 & \{S, T, V, W, X\}.
\end{align*}

Each quantizing function is offset from the previous one by one resolution ele-
ment. For every possible value of $s_i$ there exists a unique set $m_i^*$ consisting of the set of values produced by the K quantizing
functions. For example (in figure 8), the value $s_1 = 7$ maps into the set $m_1^* = \{B, H,
P, V\}$.

A similar mapping is also performed on $s_2$ by the set of quantizing functions:

$2C_1 = \{a, b, c, d, e\}$
$2C_2 = \{f, g, h, i, k\}$
$2C_3 = \{m, n, p, q, r\}$
$2C_4 = \{s, t, v, w, x\}$

For example, the value $s_2 = 10$ maps into the set $m_2^* = \{c, j, q, v\}$. Now, if the $s_1$
component in figure 8 corresponds to the position of the elbow joint, the mossy fiber
labeled B will be maximally active whenever

\begin{align*}
\text{Figure 6. Typical responses of mossy fibers to the sensory variable they encode.}
\end{align*}

\begin{align*}
\text{Figure 7: Three different mossy fibers encoding a single sensory variable (elbow position). All three fibers maximally active simultaneously indicates that the elbow lies between 118° and 120°.}
\end{align*}
POURLY ANNOUNCES THE NEWEST PET MICROCOMPUTERS BY COMMODORE!

The PET™ is now a truly sophisticated Business System with the announcement of these Peripherals.

THE NEW Commodore PET PERIPHERALS!

Dual Drive Floppy Disk

The Dual Drive Floppy is the latest in Disk technology with extremely large storage capability and excellent file management. As the Commodore disk is an "Intelligent" peripheral, it uses none of the RAM (user) memory of the PET™. The Floppy Disk operating system used with the PET™ computer enables a program to read or write data over the IEEE to the PET™. The Floppy Disk is a reliable low cost unit, and is convenient for high speed data transfer. Due to the latest technological advances incorporated in this disk, a total of 360K bytes are available in the two standard 5½" disks, without the problems of double tracking or double density. This is achieved by the use of two microprocessors and fifteen memory IC's built into the disk unit. Only two connections are necessary — an A/C cord and PET™ interface cord.

CALL FOR COMPLETE SPECIFICATIONS! IMMEDIATE DELIVERY!

NEW ENGLAND AREA SUB-DEALERSHIP INQUIRIES INVITED!

WE CANNOT LIST ALL OF OUR SOFTWARE AND HARDWARE PRODUCTS CALL OR WRITE FOR OUR *FREE* SOFTWARE/HARDWARE DIRECTORY

ALL NEECO PETS CARRY A FULL ONE-YEAR NEECO WARRANTY.

NEW ENGLAND ELECTRONICS CO., INC.
679 HIGHLAND AVE., NEEDHAM, MASS. 02194
MON. - FRI. 9:30 - 5:30, EST.
(617) 449-1760
MASTERCHARGE OR VISA ACCEPTED
TELEX NUMBER 851021, NEECO

LARGE TYPEWRITER KEYBOARDS NOW AVAILABLE!

2040 Floppy Drive

PET 2001 — 32K

NEW ENGLAND ELECTRONICS co., INC.
679 HIGHLAND AVE., NEEDHAM, MASS. 02194
MON. - FRI. 9:30 - 5:30, EST.
THE MINIMAX SERIES WAS DESIGNED TO OFFER THE MARKET MINICOMPUTER CAPABILITIES AT MICROCOMPUTER PRICES. COMPARE THE CAPABILITIES & PRICE!

CONTACT NEECO FOR FULL SPECS - FREE MINIMAX MANUAL.

THE MINIMAX SERIES COMPUTER WAS DESIGNED BY INDUSTRY PROFESSIONALS. COMPARE THE PRICE AND FEATURES TO ANY OTHER COMPUTER IN ITS CLASS!

• THE MINIMAX SERIES COMPUTER IS AN INTEGRATED, COMPACT UNIT CONTAINING THE CPU, DUAL DENSITY DISK STORAGE, 12 INCH CRT, AND FULL STYLE KEYBOARD, WITH SEPARATE NUMERIC ENTRY PAD. ALL KEYS (INCLUDING CURSOR) WITH FULL REPEAT.
• HYBRID 3 MEGAHERTZ 6502 CPU • 10K SYSTEM RAM (4K USER)
• FASTEST FLOPPY DISK ACCESS (24K LOADS IN 4.2 SECONDS)
• 16K ROM CONTAINS COMPUTHINK BASIC (AN EXTENDED MICROSOFT BASIC) WITH EXTENDED PRECISION, DOS INCLUDES COMPLETE FILE I/O WITH FULL RANDOM ACCESS, COMPLETE MONITOR WITH DEBUG & TRACE, AND TINY 6502 ASSEMBLER
• COMPLETE HIGH RESOLUTION GRAPHICS WITH INDIVIDUAL DOT (240x512) POINT SCREEN ADDRESSABILITY
• FULL SCREEN TEXT EDITING WITH OVERWRITE, INSERTION OR DELETION
• INDIVIDUAL FIELD EDITING WITH FIELD PROTECT AND AUTO SKIP TO NEXT FIELD
• DISK STORAGE SYSTEM TRANSFERS 6K PER SECOND WITH AUTO VERIFY AND PARITY CHECK
• 12 INCH CRT-64 CHARACTERS BY 30 LINES. UP TO THREE PROGRAMMABLE CHARACTER FONTS FOR LANGUAGES OR SPECIAL CHARACTERS
• SWITCHABLE 110 OR 220V OPERATION
• HYBRID CPU IS MICROPROGRAMMABLE WITH 64 USER DEFINABLE OPCODES, CHOICE OF 800X OR 2.4 MEGABYTE DISK STORAGE
• FULL SERIAL RS-232C PORT WITH PROGRAMMABLE BAUD RATES AND MODEM CONTROL SIGNAL
• DEDICATED DISK PORT
• PRINTER PORT SUPPORTS PARALLEL COMMERCIAL PRINTERS
• 24 PIN I/O USER PORT

MINIMAX I - .8 MEGABYTE ON LINE MINIFLOPPY STORAGE
MINIMAX II - 2.4 MEGABYTE ON LINE 8" FLOPPY STORAGE

MINIMAX I - $4495
MINIMAX II - $5995

PRINTER NOT INCLUDED IN PRICE

SPECIAL DEALER PRICING AVAILABLE ON DEMONSTRATION MINIMAX AND SOFTWARE PACKAGES TO QUALIFIED SELECTED DEALERS. CONTACT NEECO FOR INFO.

THE MINIMAX WAS DESIGNED AND IS MANUFACTURED BY COMPUTHINK COMPUTER CORP. DISTRIBUTED IN EUROPE AND THE EASTERN U.S. BY NEECO.

IN ADDITION TO HARDWARE CAPABILITIES THAT ARE UNMATCHED IN THE INDUSTRY, THE MINIMAX COMPUTER SUPPORTS A COMPLETE DATA BASE SYSTEM (PAGEMATE), CONTAINING FULL STATISTICAL, SORTING, AND EDIT FUNCTIONS. A PLM COMPILER IS AVAILABLE, FULL BUSINESS SUPPORT SOFTWARE IS AVAILABLE AND MORE—FULL DEALER SUPPORT IS AN IMPORTANT PART OF OUR MARKETING. CONTACT NEECO FOR FURTHER INFORMATION.

THE MINIMAX IS NOW AVAILABLE FOR QUANTITY DELIVERY. DOMESTIC DEALERSHIPS AVAILABLE. EUROPEAN DISTRIBUTORSHIPS/DEALERSHIPS AVAILABLE TO QUALIFYING COMPANIES WITH SUPPORT CAPABILITIES. SOFTWARE HOUSES AND OEM INQUIRIES INVITED. CONTACT NEECO.

NEW ENGLAND ELECTRONICS CO., INC.
679 HIGHLAND AVE., NEEDHAM, MA 02194
(617) 449-1760
MON-FRI, 9:00-5:30, E.S.T.
TELEX 951021, ANSWERBACK "NEECO"
Text continued from page 61:
the elbow is between 4 and 7, and less than maximally active whenever the elbow position is outside that region. Similarly, the mossy fiber labeled H is maximally active when the elbow is between 5 and 8, the fiber P maximally active between 6 and 9, and V between 7 and 10, etc. The combination of mossy fibers in the set \( m_1^* = \{B, H, P, V\} \) thus indicates that the variable \( s_1 = 7 \). If \( s_1 \) changes one position from (from 7 to 8, for example), the mossy fiber labeled B will drop out of the maximally active set \( m_1^* \) to be replaced by another, labeled C.

Encoding Advantages

This encoding scheme has a number of advantages. The most obvious is that a single precise variable can be transmitted reliably over a multitude of imprecise information channels. The resolution (or information content) of the transmitted variable depends on the number of channels. The more mossy fibers dedicated to a particular variable, the greater the precision with which it is represented.

A second equally important result is that small changes in the value of the input variable \( s_i \) have no effect on most of the elements in \( m_1^* \). This leads to a property known as generalization, which is crucial for learning and recall in a world where no two situations are ever exactly the same. In CMAC the extent of the neighborhood of generalization along each variable axis depends on the resolution of the CMAC quantizing functions. In the brain this corresponds to the width of the maximally active region of the mossy fibers.

Figure 8. A simple two variable CMAC with four quantizing functions on each variable. A detailed explanation is in the text.
NOW YOU CAN AFFORD
THE WORLD FAMOUS
Diablo Printer Terminal

IPS | PROUDLY ANNOUNCES A PRICE
BREAKTHROUGH ON DAISY WHEEL
PRINTER TERMINALS

Featuring: RS 232 compatibility at 110, 150, 300, and 1200 baud: Print speed of 45 char/sec: Graphics capability: Over 30 print wheels from 10 to 15 char/inch: 5 ribbons in color, cloth, film or two color: Form length adjustable: Spacing adjustable from 120 pitch to .1 pitch: Line spacing adjustable from 48 lines/in. to .4 line/in.: Identical in every respect to its high priced sister the Diablo HyTerm II* model 1620.

Price: $2,595.00 each in single units, quantity discounts available.

Service: Major cities in the U.S. and Canada.

You have probably waited a long time to purchase your computer printer at a reasonable price. Wait no longer! The Diablo 1620 lists for $3,240.00 we sell the IPSI 1622 for $2,595.00; save well over $600.00. For immediate action phone in your order collect and we will ship your terminal the same day.

We also Stock Diablo 1345A (Hytype II*) mechanisms for immediate shipment at the low price of $1,150.00. Order one now and build your own terminal using your own CPU, or do-it-yourself by buying our 1622 kit and assemble your own terminal within 10 hours for $2,295.00

If you've always wanted to turn your computer onto word processing, this is the only way to go. Ask for our free word processing software.

IPS | INTERNATIONAL PERIPHERAL SYSTEMS, INC.
1849 N. HELM
FRESNO, CALIFORNIA 93727
CALL (209) 252-3635

*Hyterm and HyType are registered trademarks of Xerox Corp.
The number of mossy fibers dedicated to a variable determines precision of its representation.

M → A Mapping

Just as we can identify (or name) mossy fibers by the input variables they encode, so we can identify granule cells by the mossy fibers which provide them with input. Each granule cell receives input from several different mossy fibers, and no two granule cells receive input from the same combination of mossy fibers. This means that we can compute a unique name (or address) for each granule cell by simply listing the mossy fibers which contact it. For example, a granule cell contacted by two mossy fibers B and c can be named (or addressed) Bc.

In the CMAC example in figure 8, 25 granule cells are identified by their contacts with mossy fibers from the quantizing functions \(1C_1\) and \(2C_1\). 25 other granule cells are identified by \(1C_2\) and \(2C_2\), 25 by \(1C_3\) and \(2C_3\), and 25 more by \(1C_4\) and \(2C_4\). There are, of course, many other possible combinations of mossy fiber names which might be used to identify a much larger number of granule cells. For this simple example, however, we will limit our selection to the permutation of corresponding quantizing functions along each of the coordinate axes. This provides a large and representative sample which uniformly spans the input space. Furthermore, this particular naming algorithm is simple to implement either in software or hardware.

We can define A to be the set of all granule cells identified by their mossy fiber inputs. Of course, all of the granule cells in A are not active at the same time. As was previously noted, most granule cells are inhibited from firing by Golgi cell gain control feedback. Only the small percentage of granule cells whose input mossy fibers are all maximally active can rise above threshold. We will define the set of active granule cells as \(A^*\).

Since we already know which mossy fibers are maximally active (ie: those mossy fibers in the sets \(m_1^*\)), we can compute names of granule cells in \(A^*\). For example, in figures 8 and 10, if \(s_1 = 7\) and \(s_2 = 10\), then \(m_1^* = \{B, H, P, V\}\) and \(m_2^* = \{c, i, j, q, v\}\). The active granule cells in \(A^*\) can now be computed directly as \(A^* = \{Bc, Hj, Pq, Vv\}\). All other granule cell names in the larger set A involve at least one mossy fiber which is not maximally active (ie: not in \(m_1^*\) or \(m_2^*\)).

Note that, as illustrated in figure 9, the granule cell Bc will be active as long as the input vector remains in the region of input space \(4 \leq s_1 \leq 7\) and \(8 \leq s_2 \leq 11\). Thus, the generalizing property introduced by the \(S \rightarrow M\) mapping carries through to the naming of active granule cells. A particular granule cell is active whenever the input vector \(S\) lies within some extended region, or neighborhood, of input space. Other granule cells are active over other neighborhoods. These neighborhoods overlap, but each is offset from the others so that for any particular input \(S\), the neighborhoods in \(A^*\) all overlap at only one point, namely the point defined by the input vector. This is illustrated in figure 10. If the input vector moves one resolution element in any direction, for example, from \((7, 10)\) to \((8, 10)\), one active granule cell (Bc) drops out of \(A^*\) to be replaced by another (Cc).

A → \(p\) Mapping

Granule cells give rise to parallel fibers which act through weighted connections on the Purkinje output cell, varying its firing rate. Each cell in A is associated with a weight which may be positive or negative. Only the cells in \(A^*\) have any effect on the Purkinje output cell. Thus, the Purkinje output sums only the weights selected (or addressed) by \(A^*\). This sum is the CMAC output scalar variable \(p\). For example, in figure 8, \(S = (7, 10)\) maps into \(A^* = \{Bc, Hj, Pq, Vv\}\) which selects the weights:

\[
\begin{align*}
W_{Bc} &= 1.0 \\
W_{Hj} &= 2.0 \\
W_{Pq} &= 1.0 \\
W_{Vv} &= 0.0.
\end{align*}
\]
In an effort to offer products that meet the continually changing demands of the microcomputer industry, TARBEIII ELECTRONICS is pleased to offer immediate delivery of these quality components and operating software. All TARBEIII products are available from computer store dealers everywhere.

**Tarbell Floppy Disk Interface**

- Plugs directly into your IMSAI or ALTAIR and handles up to 4 standard single drives in daisy chain.
- Operates at standard 230K per second on normal disk format capacity of 230K bytes.
- Works with modified CP/M Operating System and BASIC-E Compiler.
- Hardware includes 4 extra IC slots, built-in phantom bootstrap and onboard crystal clock. Uses WD 1771 LSI chip.
- Full 6-month warranty and extensive documentation.
- Kit $190 . Assembled $265.

**Tarbell 32K RAM Memory**

- 32K Static Memory
- S-100 Bus Connector
- 9 regulators provide excellent heat distribution.
- Extended addressing (bank switching).
- Phantom line.
- Low power requirement.
- 20-Page operating manual.
- Full 1-year warranty.
- Assembled and tested full price only $625
- 16K version also available, assembled and tested only $390.

**Tarbell Cassette Interface**

- Plugs directly into your IMSAI or ALTAIR.
- Fastest transfer rate: 187 (standard) to 540 bytes/second.
- Extremely reliable—Phase encoded (self-clocking).
- 4 extra status lines, and 4 extra control lines.
- 37-page manual included.
- Device code selectable by DIP-switch.
- Capable of generating Kansas City tapes.
- No modification required on audio cassette recorder.
- Complete kit $120 . Assembled $175.
- Manual may be purchased separately $8.
- Full 6-month warranty on kit and assembled units.

**Tarbell Disk BASIC**

- Runs on 8080, 8085 or Z80
- Searches a file quickly for a string.
- Up to 64 files open at once.
- Random Access.
- Assignment of I/O.
- Alphanumeric line labels allowed.
- Read and Write string or numeric data.
- Unlimited length of variable names and strings.
- Procedures with independent variables.
- Number system 10 digits BCD integer or floating point.
- Chain to another program.
- Cause programs to be appended onto programs already in memory.
- Cause interpreter to enter edit mode using 15 single character edit commands.
- Occupies 24K of RAM. Tarbell BASIC on CP/M Disk ............ $48.
- Source on paper or CP/M Disk .. $25.
- CP/M and BASIC-E on disk with manuals ........ $100.

Specify drive for assembled units. Complete disk subsystems with operating software available. Please inquire for details.

**Tarbell Cassette BASIC**

Includes most features of ALTAIR Extended BASIC, plus these added features:
- Assignment of I/O.
- Alphanumeric line labels.
- Unlimited length of variable names and strings.
- Number system 10 digits BCD integer or floating point.
- Procedures with independent variables.
- Read and Write string data.
- Multi-file capability.

Full price with complete documentation ............. $48.
Prepaid, COD, or cash only. California residents please add 6% sales tax.

ALTAIR is a trademark/tradename of Pertec Computer Corporation
CP/M is a trademark/tradename of Digital Research

---

Circle 360 on inquiry card.
These weights are summed to produce the output:

\[ p = 4.0. \]

Thus the input \( S = (7, 10) \) produces the output \( h(S) = 4 \).

In figure 8 four weights are selected for every \( S \) vector in input space. Their sum is the value of the output \( p \). As the input vector moves from any point in input space to an adjacent point one weight drops out to be replaced by another. The difference in value of the new weight minus the old is the difference in value of the output at the two adjacent points. Thus, the difference in adjacent weights is the partial derivative (or partial difference) of the function at that point. As the input vector \( S \) moves over the input space, a value \( p \) is output at each point. We can therefore say that the CMAC computes the function:

\[ p = h(S). \]

The particular function \( h \) computed depends on the particular set of values stored in the table of weights. For example, the set of weights shown in figure 8 computes the function shown in figure 11.

In the cerebellum there are many Purkinje cells which receive input from essentially the same mossy fibers. Thus, there are many CMACs all computing on the same input vector \( S \). We can therefore say that a set of \( L \) CMACs computing on the same input vector produces a vector mapping:

\[ P = H(S). \]

Data Storage in CMAC

One of the most fascinating, intensively studied, and least understood features of the brain is memory, and how data is stored in memory. In the cerebellum each Purkinje cell has a unique fiber, a climbing fiber, which is believed to be related to learning. Fibers from an area called the locus coeruleus have recently been discovered which appear to be related to learning. In addition, a number of hormones have been shown to have profound effects on learning and retention of learned experiences.

While the exact mechanism (or mechanisms) for memory storage are as yet unknown, the cerebellar model upon which CMAC is based hypothesizes that climbing fibers carry error correction information which “punishes” synapses that participate in erroneous firing of the Purkinje cell. The amount of error correction that occurs at any one experience may depend on factors such as the state of arousal or emotional importance attached by the brain's evaluation centers to the data being stored during the learning process.

Cerebellar learning is modeled in CMAC by the following procedure:

- Assume that \( \hat{H} \) is the function we want CMAC to compute. Then \( \hat{P} = H(S) \) is the desired value of the output vector for each point in the input space.
- Select a point \( S \) in input space where \( \hat{P} \) is to be stored. Compute the current value of the function at that point \( P = H(S) \).
- For every element in:
  
  \[ p_1, p_2, \ldots, p_L \]

  and in:
  
  \[ \hat{p}_1, \hat{p}_2, \ldots, \hat{p}_L \]

  if:
  
  \[ |\hat{p}_i - p_i| < \xi_i \]

  where \( \xi \) is an acceptable error, then do nothing; the desired value is already stored. (\( |\hat{p}_i - p_i| \) is the absolute value of \( \hat{p}_i - p_i \).) However, if \( |\hat{p}_i - p_i| > \xi_i \) then add \( \Delta_i \)

---

**Figure 10.** The input vector \((s_1, s_2) = (7, 10)\) selects weights \(Bc, Hj, Pq, \) and \(Vv\). These all overlap only at the point \((7, 10)\). If the input vector \((s_1, s_2)\) moves to \((8, 10)\) the weight \(Bc\) will drop out to be replaced by \(Cc\).
to every weight which was summed to produce $p_i$ where:

$$\Delta_i = g \left( \frac{\hat{p}_i - p_i}{|A^*|} \right)$$

(1)

$|A^*|$ is the number of weights in the set $A^*$ which contributed to $p$, and $g$ is a gain factor which controls the amount of error correction produced by one learning experience.

If $g = 1$, then CMAC produces oneshot learning which fully corrects the observed error in one data storage operation. If $0 < g < 1$, then each learning experience moves the output $p_i$ only in the direction of the desired value $\hat{p}_i$. More than one memory storage operation is then required to achieve correct performance.

An example of how an arbitrary function such as:

$$\hat{p} = (\sin x)(\sin y)$$

where:

$$x = 2\pi s_1/360$$

and:

$$y = 2\pi s_1/360$$

Figure 11. The particular set of weights shown in Figure 8 will compute the function shown here.
Figure 12. The effect of training CMAC on the function \( \hat{p} = \sin \left( \frac{2\pi}{360} s_1 \right) \sin \left( \frac{2\pi}{360} s_2 \right) \). (a) One training at \((s_1, s_2) = (90, 90)\). (b) A second training at \((s_1, s_2) = (270, 90)\). (c) Training at 16 points along a trajectory defined by \( s_1 = 90 \). (d) Training at 175 selected points scattered over the input space.

can be stored in CMAC is shown in figure 12. In this example the input is defined with unity resolution over the space \( 0 < s_1 < 360 \) and \( 0 < s_2 < 180 \), and the number of weights selected by each input is \(|A^*| = 32\).

Initially all the weights were equal to 0. The point \( S_1 = (90, 90) \) was chosen for the first data entry. The value of the desired function \( \hat{p} = h(90, 90) \) is 1. By formula (1) (where \( g = 1 \)) each of the weights selected by \( S = (90, 90) \) is set to 1/32, causing the proper value to be stored at \( S = (90,90) \) as shown in figure 12a. After two data storage operations, one at \( (90, 90) \), the other at \( (270, 90) \), the contents of the CMAC memory are as shown in figure 12b. After 16 storage operations along the \( s_2 = 90 \) axis the results are as shown in figure 12c. After 175 storage operations scattered over the entire input space, the contents of the CMAC memory are as shown in figure 12d.

CMAC Memory Requirements

The CMAC \( S \rightarrow A^* \) mapping corresponds to an address decoder wherein \( S \) is the input address and the active granule cells in \( A^* \) are select lines. These access weights whose sum can be interpreted as the contents of the address \( S \). In a conventional memory, each possible input address selects a unique single location wherein is stored the contents of that address, as illustrated in figure 13a. In CMAC each possible input address selects a unique set of memory locations, the sum of whose contents is the contents of the input address, as shown in figure 13b.

This suggests that the Cerebellar Model...
Arithmetic Computer might require considerably less memory than a conventional lookup table in storing certain functions. The reason is that the number of ways that \( x \) elements can be selected from a table of \( y \) entries always exceeds \( y \) and, in some cases, it does so by orders of magnitude.

A conventional memory requires \( R^N \) memory locations to store a function of \( N \) variables, where \( R \) is the number of resolution elements on each variable. CMAC requires at most \( K \times Q^N \) memory locations, when \( K \) is the number of quantizing functions and \( Q \) the number of resolution elements on each quantizing function.

A modest example of CMACs reduced memory requirements can be seen in figure 8 where \( N = 2 \) and \( R = 17 \). Here then are \( 17^2 \), or 289, possible input vectors. The CMAC shown has only 100 weights since \( K = 4 \) and \( Q = 5 \). Thus \( K \times Q^N = 100 \). This savings in memory size becomes increasingly significant for large \( N \). It allows CMAC to store a large class of low resolution functions of up to 12 variables over the entire input space with computer memory of practical size (less than 100 K bytes), whereas conventional table lookup becomes impractical for similar functions of more than four variables.

An even greater savings in memory requirements can be achieved by the use of hash coding techniques in the selection of addresses for the elements in \( A^\ast \). Hash coding allows CMAC to store functions of many variables, so long as the information content of the portion of the function stored does not exceed the number of bits in the CMAC.
memory. For example in figure 12, the 360 by 180 (over 64,000) element input space is represented in a 1024 location CMAC memory by hash coding.

Hash coding is a commonly used memory addressing technique for compressing a large but sparsely populated address space into a smaller, more densely populated one. (See "Making Hash with Tables" by Terry Dolhoff in Programming Techniques: Program Design, BYTE Books, 1979.) Many addresses in the larger space are mapped

Figure 13: (a) In a conventional memory, storage of a function of N variables with resolution R on each input variable requires R^N memory locations. Figure 13b (page 74) illustrates the CMAC distributed memory look-up technique.
THE SEARCH FOR A SMALL COMPUTER SYSTEM STARTS HERE

It's the 3rd Annual National Small Computer Show
New York Coliseum, August 23-26, 1979

presenting the state-of-the-art showcase for micro- and mini-systems technology and software. Here you can survey virtually all makes and models of small computers, whether your interest runs to a no-nonsense micro priced in the hundreds of dollars or a powerful mini costing $20,000 or more. They're all here.

The world of small computers is quite large, extending to business and professional offices, scientific research, medicine and bionics, education, the home and hobbyist, therapeutic applications for the handicapped, design and engineering. A full selection of lectures is presented to provide a grasp of small systems technology, so that you know what to consider when buying a computer or word processor. It's the first step in discovering what a system can really do for you!

NSCS lectures include sessions on system selection, computer languages, word processing functions, artificial intelligence, software applications, and a dozen more topics for people of all interests.

Plan now to attend. There will be about 30,000 square feet of exhibits, and more than 40 hours of lectures from which to choose. Registration fee is only $5.00 per day, including lectures.

LECTURES: (Program subject to change)

Thursday, August 23
1 p.m. The Peril of Becoming a Machine-Oriented Business User
1 p.m. Introduction to Small Business Systems
2 p.m. Selecting a Word Processing System
2 p.m. Distributed Data Processing
3 p.m. Accounts Receivable/General Ledger/Accounts Payable
3 p.m. Is There a Computer in Your Educational Future
4 p.m. Mailing Lists: Load, Time and Cost
4 p.m. Word Processing Systems in the Law Office
5 p.m. Basic BASIC
5 p.m. Achieving Quality Control in Word Processing

Friday, August 24
1 p.m. The Peril of Becoming a Machine-Oriented Business User
1 p.m. Introduction to Small Business Users
2 p.m. Selecting a Word Processing System
2 p.m. Distributed Data Processing
3 p.m. Accounts Receivable/General Ledger/Accounts Payable
3 p.m. How to Write a User-Oriented Program
4 p.m. Efficient Expansion of a Small System
4 p.m. Investment Analysis
5 p.m. Accounts Receivable/General Ledger/Accounts Payable
5 p.m. Exploiting the Apple/Dow Jones Computer Link

Saturday, August 25
11 a.m. Introduction to Personal Computing
11 a.m. Unassigned at press time
12 p.m. Computer Music Update
12 p.m. Unassigned at press time
1 p.m. Introduction to PASCAL
1 p.m. Computer Art Forms
2 p.m. Household Applications
2 p.m. Artificial Intelligence
3 p.m. How to Write a User-Oriented Program
3 p.m. Investment Analysis
4 p.m. Basic BASIC
4 p.m. Unassigned at press time

Sunday, August 26
11 a.m. Introduction to Personal Computing
11 a.m. Computer Music Update
12 p.m. Household Applications
12 p.m. Unassigned at press time
1 p.m. Efficient Expansion of a Small System
1 p.m. Computer Art Forms
2 p.m. Unassigned at press time
2 p.m. Unassigned at press time
3 p.m. Microcomputers for the Handicapped: Update
3 p.m. Exploiting the Apple/Dow Jones Computer Link
4 p.m. Mailing Lists: Load, Time and Cost
4 p.m. Introduction to PASCAL

REGISTRATION FOR AMERICA'S BIGGEST SMALL COMPUTER SHOW
Please register me for the 3rd Annual National Small Computer Show, Aug. 23-26, 1979.

NAME ____________________________ BUSINESS TITLE (If Any) ____________________________
COMPANY (If Any) __________________________________ TELEPHONE ____________________
ADDRESS __________________________________ ZIP ______

(Complete main interest)
☐ Banking/Insurance
☐ Business office
☐ Communications
☐ Educational
☐ Government
☐ Hobby
☐ Industrial/Manufacturing
☐ Military
☐ Professional
☐ Stock Brokerage
☐ Transportation
☐ Utility
☐ Wholesale/Retail
☐ Other

(Choose main job function)
☐ Accountant
☐ Administrator (Business)
☐ Architect/Builder
☐ Art Director
☐ Banker
☐ Computer technician
☐ Consultant
☐ Controller
☐ Engineer
☐ Industrial Designer
☐ Lawyer/new office mgr.
☐ Marketing manager
☐ Medical doctor
☐ Medical technician
☐ Office manager
☐ Programmer
☐ Public Servant
☐ Research/Development
☐ Teacher
☐ DP manager
☐ WP manager
☐ WP operator
☐ Stock Broker
☐ Systems Analyst
☐ Student
☐ Other

☐ ONE DAY $5
☐ TWO DAYS $10
☐ THREE DAYS $15
☐ FOUR DAYS $20

Mail with payment of $5 for each day you wish to attend. Use one form per person. Registration badge will be sent by mail in early August. Check or money order only.

Mail prior to Aug. 10 to:
National Small Computer Show,
110 Charlotte Place,
Englewood Cliffs, N.J. 07632

Circle 258 on inquiry card.
In a CMAC model, each input selects a unique set of memory locations. The number of unique sets which can be selected from $M$ locations is much larger than the input $M$.

In CMAC, hashing noise is randomly scattered over the input space each time new data is stored. Thus each new data storage operation degrades previously stored data somewhat. The effect is that the contents of a CMAC memory are most accurately defined in the regions where it is most recently stored. Old data tends to gradually fade, or be “forgotten”, due to being hashed over.

**CMAC Memory Generalization**

The fact that each possible CMAC input vector selects a unique set of memory locations rather than a single location implies that any particular location may be selected by more than one input vector. In fact, the
Color. VP-590 add-on Color Board allows program control of 8 brilliant colors for graphics, color games. Plus 4 selectable background colors. Includes sockets for 2 auxiliary keypads (VP-580). $69.*

Sound. VP-595 Simple Sound Board provides 256 tone frequencies. Great for supplementing graphics with sound effects or music. Set tone and duration with easy instructions. $24.*

Music. VP-550 Super Sound Board turns your VIP into a music synthesizer. 2 sound channels. Program control of frequency, time and amplitude envelope (voice) independently in each channel. Program directly from sheet music! Sync provision for controlling multiple VIPs, multitrack recording or other synthesizers. $49.*

Memory. VP-570 RAM Expansion Board adds 4K bytes of memory. Jumper locates RAM in any 4K block of up to 32K of memory. On-board memory protect switch. $95.*

EPROM Programmer. VP-565 EPROM Programmer Board comes complete with software to program, copy and verify 5-volt 2716 EPROMs—comparable to units costing much more than the VP-565 and VIP put together! Programming voltages generated on board. ZIF PROM socket included. $99.*

EPROM Interface. VP-560 EPROM Interface Board locates two 5-volt 2716 EPROMs (4K bytes total) anywhere in 32K of memory. VIP RAM can be re-allocated. $34.*

ASCII Keyboard** Fully encoded, 128-character ASCII encoded alpha-numeric keyboard. 58 light touch keys including 2 user defined keys! Selectable upper and lower case. Handsomely styled. Under $50.*

COSMAC VIP lets you add computer power a board at a time.

With these new easy-to-buy options, the versatile RCA COSMAC VIP (CDP18S711) means even more excitement. More challenges in graphics, games and control functions. For everyone, from youngster to serious hobbyist. And the basic VIP computer system starts at just $249* assembled and ready to operate.

Simple but powerful—not just a toy.
Built around an RCA COSMAC microprocessor, the VIP includes 2K of RAM. ROM monitor. Audio tone with a built-in speaker. Plus 8-bit input and 8-bit output port to interface relays, sensors or other peripherals. It's easy to program and operate. Powerful CHIP-8 interpretive language gets you into programming the first evening. Complete documentation provided.

Take the first step now.
Check your local computer store or electronics parts house. Or contact RCA VIP Marketing, New Holland Avenue, Lancaster, PA 17604. Phone (717) 291-5848.

*Suggested retail price. CDP18S711 does not include video monitor or cassette recorder.
**Available 1st Quarter, 1979.

The fun way into computers.

RCA VIP Marketing, New Holland Avenue, Lancaster, PA 17604. Phone (717) 291-5848.

See the RCA VIP at the 3rd Annual National Small Computer Show at the New York Coliseum, August 23-26, booth 4211.

Circle 322 on inquiry card.
Figure 14. The CMAC memory generalizes. (a) $S_2$ selects three out of four of the same weights as $S_1$. Thus output $h(S_2)$ will be similar to $h(S_1)$, differing only by the contents of the location not in common. (b) When $S_2$ is outside of the neighborhood of generalization of $S_1$, the overlap goes to 0 (except for random hashing collisions).

$S \rightarrow A^*$ mapping insures that any two input vectors which are similar (i.e., close together in input space) will activate many of the same granule cells, and hence select many of the same weights. This is the property of CMAC which causes it to generalize.

In figure 14a the input vector $S_2$ selects three out of four of the same memory locations as $S_1$. Thus, the output $h(S_2)$ will be similar to $h(S_1)$, differing only by the contents of the single location which is not in common. The $S \rightarrow A^*$ mapping controls the amount of overlap between sets of selected memory locations such that, as the input space distance between two input vectors increases, the amount of overlap decreases. Finally, at some distance the overlap becomes 0 (except for random hashing collisions), as in figure 14b, and the sets of selected memory locations are disjoint. At that point input $S_2$ can be said to be outside the neighborhood of generalization of $S_1$. The value of the output $h(S_2)$ is thus independent of $h(S_1)$.

The extent of the neighborhood of generalization depends on both the number of elements in the set $A^*$ and the resolution of the $S \rightarrow m_i^*$ mappings. It is possible in CMAC to make the neighborhood of generalization broad along some variable axes and limited along others by using different resolution quantizing functions for different input variables. This corresponds to the effect in the cerebellum where some input variables are resolved finely by many mossy fibers and others resolved more coarsely by fewer mossy fibers.

A good example of generalization can be
seen in figure 12a. Following a single data storage operation at $S_1 = (90, 90)$ we find that an input vector $S_2 = (91, 90)$ will produce the output $p = 31/32$ even though nothing had ever been explicitly stored at $(91, 90)$. This occurs because $S_2$ selects 31 of the same weights as $S_1$. A third vector $S_3 = (92, 90)$ or a fourth $S_4 = (90, 92)$, will produce $p = 30/32$ because of sharing 30 weights with $S_1$. Not until two input vectors are more than 32 resolution elements apart do they map into disjoint sets of weights.

As a result of generalization, CMAC memory addresses in the same neighborhood are not independent. Data storage at any point alters the values stored at neighboring points. Pulling one point to a particular value as in figure 12a produces the effect of stretching a rubber sheet.

Generalization has the advantage that data storage (or training) is not required at every point in the input space in order for an approximately correct response to be obtained. This means that a good first approximation to the correct $H$ function can be stored for a sizable envelope around a $T_s$ trajectory by training at only a few points along that trajectory. For example, figure 12c demonstrates that training at only 16 points along the trajectory defined by $s_2 = 90$ generalizes to approximately the correct function for all 360 points along that trajectory plus a great many more points in an envelope around that trajectory. Further training at 175 points scattered over the entire space generalizes to approximately the correct response for all 360 by 180 (over 64,000) points in the input space as shown in figure 12d.

Generalization enables CMAC to predict
Low Cost Add-On Storage for Your TRS-80®
In the Size You Want.

When you’re ready for add-on disk storage, we’re ready for you.

Ready with six mini-disk storage systems — 102K bytes to 591K bytes of additional on-line storage for your TRS-80®.

• Choose either 40-track TFD-100™ drives or 77-track TFD-200™ drives.
• One-, two- and three-drive systems immediately available.
• Systems include Percom PATCH PAK #1™, on disk, at no extra charge. PATCH PAK #1™ de-glitches and upgrades TRSDOS® for 40- and 77-track operation.
• TFD-100™ drives accommodate "flippy disks." Store 205K bytes per mini-disk.
• Low prices. A single-drive TFD-100™ costs just $399. Price includes PATCH PAK #1™ disk.
• Enclosures are finished in system-compatible "Tandy-silver" enamel.

Whether you need a single, 40-track TFD-100™ add-on or a three-drive add-on with 77-track TFD-200™s, you get more data storage for less money from Percom.

Our TFD-100™ drive, for example, lets you store 102.4K bytes of data on one side of a disk — compared to 80K bytes on a TRS-80® mini-disk drive — and 102.4K bytes on the other side, too. Something you can’t do with a TRS-80® drive. That’s almost 205K bytes per mini-disk.

And the TFD-200™ drives provide 197K bytes of on-line storage per drive — 197K, 394K and 591K bytes for one-, two- and three-drive systems.

PATCH PAK #1™, our upgrade program for your TRSDOS®, not only extends TRSDOS® to accommodate 40- and 77-track drives, it enhances TRSDOS® in other ways as well. PATCH PAK #1™ is supplied with each drive system at no additional charge.

The reason you get more for less from Percom is simple. Peripherals are not a sideline at Percom. Selling disk systems and other peripherals is our main business — the reason you get more engineering, more reliability and more back up support for less money.

In the Product Development Queue... a printer interface for using your TRS-80® with any serial printer, and... the Electric Crayon™ to map your computer memory onto your color TV screen — for games, animated shows, business displays, graphs, etc. Coming PDQ!

From PERCOM
One-Drive System:
$399. (40-track) & $575. (77-track)
Two-Drive System:
$795. (40-track drives) & $1350. (77-track drives)
Three-Drive System:
$1195. (40-track drives) & $2025. (77-track drives)
Requires Expansion Interface, Level II BASIC & 16K RAM

To order add-on mini-disk storage for your TRS-80®,
or request additional literature, call Percom's toll-free number: 1-800-527-1592. For detailed Technical Information call (214) 272-3421.

Orders may be paid by check or money order, or charged to Visa or Master Charge credit accounts. Texas residents must add 5% sales tax.

Percom 'peripherals for personal computing'

PERCOM DATA COMPANY, INC.
DEPT. B • 211 N. KIRBY • GARLAND, TX. 75042

TM TFD-100 TFD-200 PATCH PAK and Electric Crayon are trademarks of PERCOM DATA COMPANY
*TRS-80 and TRSDOS are trademarks of Tandy Corporation and Radio Shack which have no relationship to PERCOM DATA COMPANY

Percom peripherals for personal computing

78 BYTE July 1979
For your SWTP 6800 Computer...

PERCOM's™  
FLOPPY DISK SYSTEM  
the  
LFD-400

Ready to plug in and run the moment you receive it. Nothing else to buy, no extra memory. No "booting" with PerCom MINIDOS-PLUSX™, the remarkable disk operating system on EPROM. Expandable to either two or three drives. Outstanding operating, utility and application programs.

For the low $599.95 price, you not only get the disk drive, drive power supply, SS-50 bus controller/interface card, and MINIDOS-PLUSX™, you also receive:

- an attractive metal enclosure
- a fully assembled and tested interconnecting cable
- a 70-page instruction manual that includes operating instructions, schematics, service procedures and a complete listing of MINIDOS™
- technical memo updates — helpful hints which supplement the manual instructions
- a 90-day limited warranty.

SOFTWARE FOR THE LFD-400 SYSTEM

Disk operating and file management systems

INDEX™ The most advanced disk operating and file management system available for the 6800. Interrupt Driven EXeuctive 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. Multi-level file name directory includes name, extension, version, protection and date. Requires 8K RAM at S4000. Diskette includes numerous utilities...

BASIC Interpreters and Compilers

SUPER BASIC A 10K extended disk BASIC interpreter for the 6800. Faster than SWTP BASIC. Handles data files. Programs may be prepared using a text editor described below...

BASIC BANDAID™ Turn SWTP 8K BASIC into a random access data file disk BASIC. Includes many speed improvements, and program disk CHAINing...

STUBAL+™ STRUCTured BASIC Language compiler for the professional programmer. 14-dimension arrays, 2-dimension arrays. Requires 20K RAM and Linkage Editor (see below). Use of the following text editors to prepare programs. Complete with RUN-TIME and FLOATING POINT packages $249.95.

Text Editors and Processors

EDIT64 Hemenway Associates' powerful disk-based text editor. May be used to create programs and data files. Supports MACROS which perform complex, repetitive editing functions. Permits text files larger than available RAM to be created and edited...

TOUCHUP™ Modifies TSC's Text Editor and Text Processor for PerCom disk operation. ROLL function permits text files larger than available RAM to be created and edited. Supplied on diskette complete with source listing...

Assemblers

PerCom 6800 SYMBOLIC ASSEMBLER Specify assembly options at time of assembly with this symbolic assembler. Source listing on diskette...

MACRO-RELOCATING ASSEMBLER Hemenway Associates' assembler for the 6800, relocatable linking object code. Supports MACROS. Permits conditional assembly...

LINKAGE EDITOR — for STRUBAL+™ and the MACRO-Relocating assembler...

CROSS REFERENCE Utility program that produces a cross-reference listing of an input source listing file...

Business Applications

GENERAL LEDGER SYSTEM Accommodates up to 250 accounts. Financial information immediately available — no sorting required. Automatically sorts account records back to source document. User-defines account number...

FULL FUNCTION MAILING LIST 700 addresses per diskette. Powerful search, sort, create and update capability...

PERCOM FINDER™ General purpose information retrieval system...

™ trademark of PERCOM Data Company, Inc
™ trademark of Hemenway Associates Company

Now! The LFD-800 and LFD-1000. Add one, two or three LFD-800 drives and store 200K bytes per drive on-line. Add one or two (dual-drive) LFD-1000 units and store 800K bytes per unit on-line. Complete with interface/controller, DOS, cable & manuals. Two-drive systems: LFD-800 — $1549; LFD-1000 — $2495.
Dr. James S. Albus worked for NASA from 1957 to 1972 designing optical and electronic subsystems for over 15 spacecraft, and for one year managed the NASA Artificial Intelligence Program. Since 1973 he has been with the National Bureau of Standards where he has received several awards for his work in advanced computer control systems for industrial robots. He has written a survey article on robot systems for Scientific American (February 1976) and his Cerebellar Model Arithmetic Computer won the Industrial Research Magazine IR-100 Award as one of the 100 most significant new products of 1975.

on the basis of a few representative learning experiences what the appropriate behavioral response should be for similar situations. This is essential in order to cope with the complexities of real world environments where identical \( T_s \) trajectories seldom, if ever, reoccur.

An example of how CMAC uses generalization to learn trajectories in a high-dimensional space is shown in figure 15. A seven degree of freedom manipulator arm was controlled by seven CMACs, one for each joint actuator, such that the output vector \( P = H(S) \) had seven components. The input vector \( S \) to each CMAC contained 18 variables corresponding to position and velocity feedback from each of the seven joints of the arm, plus four binary bits defining the Elementary Move Command. The resolution on the feedback variables was different for each of the seven CMACs, being highest resolution from the joint driven by the output \( p_1 \) and lower for other joints in inverse proportion to their distance along the arm from the controlled joint.

The desired output trajectory \( T_{P_a} \) is shown as the set of solid curves marked (a) in figure 16. This trajectory corresponds to the Elemental Movement \(<\text{SLAP}>\) which is a motion an arm might make in swatting a mosquito.

The (i) curve in figure 17 shows the learning performance with no previous learning over twenty complete \( T_{P_a} \) "slap" motions. At the beginning of each motion the arm was positioned at the correct starting point and driven from there by the \( P \) output computed by the CMAC \( H \) function. Differences between \( P \) and \( P_a \) at 20 points along the slap trajectory were corrected by formula 1 (with \( g \) set to \( 1/20 \)). Each point on the curve in figure 17 represents the sum of all the errors for all the joints during an entire slap motion. Note that learning is rapid despite the high dimensional input space in which no two \( T_s \) trajectories were ever exactly the same. This is due to CMAC's ability to generalize from a relatively small number of specific teaching experiences to a large number of similar but not identical trajectories.

The (ii) curve in figure 17 shows the learning performance on the same twenty \( T_{P_a} \) trajectories when preceded by twenty training sessions on the \( T_{P_b} \) trajectory indicated by the dotted set of curves marked (b) in figure 16. Note that performance on \( T_{P_a} \) is consistently better following prior learning on a similar trajectory \( T_{P_b} \). The learning on \( T_{P_b} \) generalizes to the similar trajectory \( T_{P_a} \).

Needless to say, predictions based on generalization are not always correct and sometimes need to be refined by further learning. The ability of CMAC to discriminate (ie: to produce different outputs for different inputs, \( (S_1 \) and \( S_2) \)) depends upon how many weights selected by \( S_1 \) are not also selected by \( S_2 \), and how different in value those weights are. If two inputs which are close together in input space are desired to produce significantly different outputs,
FOR PERSONAL COMPUTER USERS

onComputing

A new quarterly by the staff of BYTE®

This totally new publication is entertaining, informative, and uncomplicated. It is edited for the attorney, accountant, writer and other professional or business person aware of the personal computer as a tool for business, education, home entertainment, laboratory work and other applications.

Compiled and edited by the staff of BYTE, latest developments covered in onComputing will include creative uses of the small computer, books for the computer user, how and where to buy your personal computer and numerous features concerning the fascinating world of the microprocessor.

Add onComputing to your library of "must" publications. Act now, subscribe and receive four (4) issues at $8.50 for one year (U.S.); Canada and Mexico: $10.00.

onComputing

GUIDE TO PERSONAL COMPUTING

SUBSCRIBE NOW! COMPLETE AND MAIL ATTACHED POSTAGE PAID REPLY CARD

onComputing, Inc. □ 70 Main St. □ Peterborough, NH 03458 □ Dial toll free 800-258-5485 (In NH 924-7217)
Figure 16. Two similar trajectories $T_{pa}$ and $T_{pb}$ which have different starting points but the same endpoint. Both trajectories define a version of an Elemental Movement (SLAP) which was taught to the CMACs of figure 15.

then repeated training may be required to overcome the (in this case erroneous) tendency of CMAC to generalize by building up large differences in the few weights which are not in common.

In most behavioral control situations, sharp discontinuities requiring radically different outputs for highly similar inputs do not occur. Indeed most servocontrol functions have simple S shaped characteristics along each variable axis. The complexity in control computation in multivariant servo-systems typically derives from cross-products which affect the slope of the function, or produce skewness, and nonsymmetrical hills and valleys in various corners of the N dimensional space. As can be seen from figure 11 these are the type of functions CMAC can readily store, and hence compute. Nevertheless, even on smooth functions generalization may sometimes introduce errors by altering values stored at neighboring locations which were already correct. This type of error corresponds to what psychologists call learning interference, or retroactive inhibition.

For example, in the learning of the two similar trajectories in figure 16, training on $T_{pa}$ causes degradation or interference with what was previously learned on $T_{pb}$. This can be seen in figure 18 where, after 20 training sessions on $T_{pa}$, the CMAC is trained 20 sessions on $T_{pb}$. Following this the performance on $T_{pb}$ is degraded. However, the error rate on $T_{pb}$ quickly improves
SYNCHRO-SOUND
The ORIGINA\ COMPUTER PEOPLE WHO KNOW computer\ and offer EVERYTHING you need in Small Computer Systems

Compare PRICE, QUALITY, DELIVERY, SERVICE and you'll see why you don't have to look anywhere else!

TEXAS INSTRUMENTS
810 Multi-Copy
Impact Printer

ONLY $1599.00
SINGLE QUANTITY PRICE

TERMINALS
LEAR-SIEGLER

ADMS3A
ADMS31
ADMS42

HAZELTINE

1400 .
1410 .
1500 .
Mod 1 Edit .

PRINTERs
OKIDATA

SL125
110 .

CALL* for prices

TELETYP"E

43 .

CALL* for price

COMPUTERS
NORTHSTAR
Horizon II (kit) .

CALL* for price

CROMEMBCO
System 3 .

CALL* for price

SOFTWARE
Here are nine packages that enable you to make full use of the potential in your small computer system:

ACCOUNTS RECEIVABLE
INVENTORY
ORDER ENTRY
WORD PROCESSING
PAYROLL
REAL ESTATE MGT.
MEDICAL-DENTAL
MAILING PROGRAM
SHIPPING & RECEIVING
and MORE!

Microsoft Basic, Fortran, Cobol available

CALL* for prices

CALL* for prices

CALL* for price

CALL* for price

CALL* for price

CALL* for price

CALL* for price

CALL* for price

CALL* for price

CALL* for price

MORE SPECIALS

Decwriter II .

$1595.00

44 Column Printer .

$295.00

Qume Sprint 5/45

$275.00

Livermore Acoustic

Coupler ...

$249.00

Percsi 277 Dble

Density

$1395.00

Javelin 9" Monitor...

$159.95

Imsat PCS 80/15

Anadex DF 8000

$599.00

Centronics

$995.00

Micro P.1 Printer...

$495.00

We have a full staff of Programmers and Computer Consultants to design, configure and deliver a Turnkey Computer System to meet your specific requirements.

SYNCHRO-SOUND ENTERPRISES, INC.

The Computer People
195-25 Jamaica Avenue, Jamaica, New York 11423
212/468-7067 TWX 710-682-5886

Circle 355 on inquiry card.
over another 20 training sessions. Following this another 20 training sessions are conducted on $T_{Pa}$. Again degradation in $T_{Pb}$ due to learning interference occurs, but not as severely as before. Another set of 20 training sessions on $T_{Pa}$ followed by another 20 on $T_{Pb}$ shows that the amount of learning interference is declining due to the build-up of values in the few weights which are not common to both $T_{Sa}$ and $T_{Sb}$. Thus, learning interference, or retroactive inhibition, is overcome by repetition of the learning process.

CMAC as a Computer

The ability of CMAC to store and recall (and hence compute) a general class of multivarient mathematical functions of the form $P = H(S)$ demonstrates how a relatively small cluster of neurons can calculate the type of mathematical functions required for multivarient servomechanisms, coordinate transformations, conditional branches, task decomposition operators, and IF/THEN production rules. These are the types of functions that we showed in part 1. They are required for generating goal-directed behavior (i.e.: the purposive strings of behavior patterns such as running, jumping, flying, hunting, fleeing, fighting, and mating, which are routinely accomplished with apparent ease by the tiniest rodents, birds, and even insects).

In the case of multivarient servomechanisms the S vector corresponds to commands plus feedback (i.e.: $S = C + F$). For coordinate transformations the S vector contains the arguments as well as the variables in the transformation matrix.

In the case of conditional branches, one or more of the input variables in S can be used to select different regions in input space where entirely different functions are stored. Assume, for example, that in figure 12 a third variable $s_3$ had been included in the function being stored. Assume that $s_3$ is held constant at $s_3 = 0$ while storing the function $p = (\sin x)(\sin y)$. Following that, an entirely different function, say $p = 3x + 5y^2$, could be stored with $s_3$ held constant at $s_3 = 50$. Since every point in the input space for $s_3 = 0$ is outside the neighborhood of generalization of the input space for $s_3 = 50$, there would be no interference except for random hashing collisions. The stored function would then be:

$$p = (\sin x)(\sin y) \text{ if } s_3 = 0$$
$$p = 3x + 5y^2 \text{ if } s_3 = 50$$

In the interval $0 < s_3 < 50$ the function would change smoothly from $p = (\sin x)(\sin y)$ to $p = 3x + 5y^2$. Additional func-
HARDWARE FOR TRS-80®

- Pertec Disc Drives FD-200 ........ $375.00
- These are 40-track drives that are completely compatible with the TRS-80® and Radio Shack drives. 3.0 DOS included. Will allow turning Diskette over and Write on other side.
- 16K Upgrade Kit .................. $ 99.95
- Centronics TRACTOR FEED Printer ........ $1100.00
- Complete with Cable. Shipped Freight COD.
- 200ns 16K Dynamic Memory Clips for Keyboard or Expansion Interface, Lifetime Guarantee, complete ... $119.00

ORDER NOW AND SAVE
Just list the items you want and mail this convenient coupon.

SOFTWARE BY ACS

- Monitor No. 3 ..................... $ 39.95
- Complete Machine Language Monitor for TRS-80® features: Find, EDIT, Relocate, Symbolic Dump to Tape, etc.
- Trash DATA BASE Management System ........ $ 89.95
- A complete all around DBMS for business or home. Information can be stored in any fashion. Comes on Disk with Documentation.
- PLC/END .......................... $ 15.95
- Will Patch ASCII files of Basic Programs or text or DATA FILES so they may be loaded into the Disk. Version of the Electric Pencil for Editing purposes comes on Cassettes that will automatically create a Disk file of PLC/END.
- MAKE TAPE AND MAKE DISK
  for Cassette Dealers ................ $ 69.95
  These are two programs that will allow you to take any type of Program from Disk and store it on Tape for mailing purposes. When the user receives the program in the mail on cassette, it is loaded into the computer which will automatically make a Disk file of the program.
- CP/M & C BASIC for the TRS-80®
  CP/M includes: MOVCPM, STAT, PIP, Dump, DDT, ASM (8080), ED, plus 6 user manuals.
  CP/M .................. $160.00
  C BASIC 2 includes: XREF2, CBAS2, and manuals.
- G2 LEVEL III BASIC for TRS-80® Special $39.95
- TELCOM – Telecommunications for the TRS-80® $ 29.95
  Telecommunications for the TRS-80® allows one TRS-80® to communicate with another through the RS-232-C over the phone line.

Orders received by 6:00 p.m. shipped next day on Master Charge, Visa, Certified Check or Money Order. Personal Checks require 14 days to clear. No C.O.D. Collect calls not accepted. All Hardware warranted for 90 days except Radio Shack equipment which is warranted through Radio Shack. All Software is sold "AS IS" with no guarantee. Prices subject to change without notice.

AUTOMATED COMPUTER SOFTWARE SERVICE
(615) 244-2798
Division of Computer World INC.
625 Main Street • Nashville, TN 37206

Send Check or Money Order payable to:
SOFTWARE • P.O. Box 60097 • Nashville, TN 37206

<table>
<thead>
<tr>
<th>Quan.</th>
<th>Description</th>
<th>Unit Price</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

HANDLING CHARGE $1.50
TENN. RES. ADD 6% SALES TAX TOTAL

Exp. Date

Name ________________________
Address ______________________
City ____________________ State _____ Zip _____

Check
Money Order
MasterCharge
Visa
Card No. ________
Exp. Date ________
Figure 19. A CMAC with feedback directly from output to input behaves like a finite state automaton for binary inputs and outputs. It behaves like a “fuzzy state automaton” for nonbinary s and p variables.

It is possible to construct a CMAC equivalent of any finite state automaton.
FOR FAST IMPLEMENTATION OF BUSINESS APPLICATIONS

Compact Compiler requires only 20K bytes. Our compact Version 3 Compiler requires only 20K bytes but supports a powerful subset of standard COBOL plus the CIS COBOL language extensions for conversational mode working, providing full CRT screen formatting with built-in cursor control and numeric field validation.

Other features include CIS COBOL Indexed Sequential and Interactive Debug packages, linkage to run time subroutines to CHAIN programs together, PEEK & POKE memory locations and GET & PUT to special peripherals.

Forms Generator speeds program development. Forms is a powerful utility which can be used with either the Standard or Compact COBOL Compiler. Forms generates record descriptions for Data Entry and Enquiry Programs. The operator creates and edits the required data entry form on the CRT by means of a question and answer session, and then automatically creates the required COBOL source text. This text is held on diskette as COPY files which the programmer can then simply COPY into his program in the normal COBOL way.

Standard Compiler conforms to ANSI 74 Specifications. And now our ANSI 74 standard Version 4 Compiler implements level 1 of Nucleus, Table Handling, Sequential I/O, Relative I/O, Indexed I/O, Library, Segmentation and Inter Program Communication plus many level 2 features and the CIS COBOL language extensions employed in Version 3. This compiler runs in 30K bytes and is ideal for implementing or converting large systems using modular programming. Features include dynamic program loading and fast program development turnaround.

CIS COBOL products run on the 8080 or 8086 microprocessors under the CP/M operating system. They are distributed on a variety of diskette formats and have a CONFIG utility supplied as standard, which enables you to drive many different types of CRT. All are themselves written in CIS COBOL and are therefore readily transferred to different operating environments including new operating systems and new microprocessors, making CIS COBOL a very attractive proposition for OEMs.

Circle 221 on inquiry card.
unchanged throughout the lower level trajectory deviation. This, however, is quite a different mechanism from the interrupt circuitry in the normal computer where a program counter is stored so that program execution can continue after the interrupt has been serviced.

The implication here is that a set of robot control programs modeled after a CMAC hierarchy will include no DO-loops and will not be interrupt driven. Every computing module will implement a simple state mapping function of the form $P = H(S)$.

Note also that in a CMAC hierarchy, a deviation in a higher level trajectory changes the command string, and hence the program, of all the levels below it. This implies real time modification of program statements and thus makes the use of a compiler based programming language somewhat cumbersome. A robot control system modeled after a CMAC hierarchy should use some form of an interpretable language where program statements are translated into machine code at execution time. A language similar to FORTH seems ideal. (An interpretable language can, of course, be written in a compiler based language. Also, languages can be devised which are partially compiled and partially interpreted.) We will return to these and other practical issues of computing architecture for robot control at a later time.

CMAC as a Pattern Recognizer

As was discussed in part 1, any spatial pattern can be represented as a vector. For example, a picture can be represented as an array, or ordered list, of brightness or color values. A symbolic character can be represented as an ordered list of features (or arbitrary numbers, as in the ASCII convention). Any temporal pattern can be represented as a trajectory through an N-dimensional space. For example, an audio pattern is a sequence of pressure or voltage values (i.e.: a one-dimensional trajectory). A moving picture or television scene corresponds to a sequence of picture vectors (i.e.: an N-dimensional trajectory where N is the number of picture resolution elements or pixels).

The fundamental problem of pattern recognition is to name the patterns. All the patterns with the same name are in the same class. When a pattern has been given a name we say it has been recognized. For example, when the image of a familiar face falls on my retina and I say to myself "That's George," I have recognized the visual pattern by naming it.

At this point we need to introduce some new notation to clearly distinguish between
Here it is! The most complete, totally integrated, word processing system software you've ever seen on a microcomputer. WORD-START™ really shines in ease of use, with its unique, dynamically activated menu scheme that allows typists to become word processing operators in minutes instead of weeks. For only $495, your 8080/8085/Z-80 with any CRT device becomes a word processing system with the features of a Lanier, Wang, Vydec, Xerox, etc, for far less cost.

Features like ON-SCREEN VISUAL TEXT COMPOSITION, with word wrap, plus dynamic justification and remargination. Imagine being able to type without hitting RETURN, and as WORD-START™ moves you to the next line, the preceding line is redisplayed, justified to your left and right margins! Center a line with a keystroke! Set BOLDFACE and/or underline even in mid paragraph! Rejustify text to new margins, on-screen, at will!

DYNAMIC PAGINATION shows the printer page breaks during text entry, correction, or review. Dot commands control vertical page layout, CONDITIONAL PAGINATION, page HEADINGS, page NUMBERS, etc. The integrated PRINT COMMAND can print selected pages only, pause between pages for letterhead loading, and drive any CP/M* list device. WORD-START™ runs with the CP/M interfaced printer and terminal of your choice.

SPECIFICATIONS: status line showing page, line, column of cursor; variable menu suppression for experienced operators; dynamic display of text during entry or on call permits CRT preview of printed copy; full disk buffering (document sizes up to diskette capacity); multiple file/disk edits in a session; CP/M file compatible — also useful for data entry, editing programs, etc. Commands include: cursor up/down/left/right; scroll line/screenful, up/down, once/variable rate; insert/overtype; delete character/word/line left/right/entire; variable margin set/release; set/clear tab stops; justify/ragged right; block move/copy/delete; search/replace once/n times/global/selective/whole words only; write toread from additional files; set/go to any of 10 place markers. Additional support provided for CRT's with inverse video/dual intensity, line insert/delete, and erase to end of line. Dialog interface for easy installation. AND MORE!

Registered WORD-MASTER™ users can get a $100 discount from us or your participating dealer when upgrading to WORD-START™. Offer good for a limited time only.

WORD-START™ operates with any CP/M* operating system, 32K, and either a memory mapped video board or any CRT TERMINAL with cursor control and clear screen (Hazeltine, Lear-Siegler, Soroc, Microterm, Hewlett Packard, Infoton, Beehive, Processor Tech VDM, IMSAI VIO, SD Systems, Altos, Dynabyte, GNAT, etc). We ship on 8" IBM, Micropolis II, or Northstar diskette.

Other fine MicroPro Products include:

<table>
<thead>
<tr>
<th>Product</th>
<th>Software/Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word-Star™</td>
<td>$495/25</td>
</tr>
<tr>
<td>Word-Master™</td>
<td>$150/25</td>
</tr>
<tr>
<td>Tex-Writer™</td>
<td>$75/15</td>
</tr>
<tr>
<td>Super-Sort™</td>
<td></td>
</tr>
<tr>
<td>Super-Sort™ I</td>
<td>$250/25</td>
</tr>
<tr>
<td>Super-Sort™ II</td>
<td>$200/25</td>
</tr>
<tr>
<td>Super-Sort™ III</td>
<td>$150/25</td>
</tr>
</tbody>
</table>

MICROPRO INTERNATIONAL CORPORATION
1299 4th Street, San Rafael, California 94901
Telephone (415) 457-8990    Telex 340388    Dealer/OEM inquiries invited

"PROFESSIONAL QUALITY SOFTWARE YOU CAN COUNT ON, NOW"

*CPM is a trademark of Digital Research Corp.
vectors in the sensory processing hierarchy and those in the behavior-generating hierarchy. Thus we will define the input vector to a CMAC pattern recognizer as:

$$D = E + R$$

where:

$$E = (d_1, d_2, \ldots, d_i)$$

is a vector, or list, of data variables derived from sensory input from the external environment, and:

$$R = (d_{i+1}, \ldots, d_N)$$

is a vector of data variables derived from recalled experiences, or internal context. The CMAC mapping operator in the sensory processing hierarchy will be denoted $G$ and the output $Q$ such that:

$$Q = G(D)$$

We can now define a CMAC $D$ vector to represent a sensory pattern plus context such that each component $d_i$ represents a data point or feature of the pattern plus context. The existence of the $D$ vector within a particular region of space therefore corresponds to the occurrence of a particular set of features or a particular pattern in a particular context. The recognition problem is then to find a set of CMAC weights such that the $G$ function computes an output vector:

$$Q = G(D)$$

such that $Q$ is the name of the pattern plus context $D$ as shown in figure 20.

In other words $G$ can recognize the existence of a particular pattern and context (i.e., the existence of $D$ in a particular region of input space) by outputting the name $Q$. For example,

$$Q = \text{Class I} \text{ whenever } D \text{ is in Region 1}$$
$$Q = \text{Class II} \text{ whenever } D \text{ is in Region 2}$$
$$\vdots$$
$$\vdots$$

etc.

The $D \rightarrow A$ mapping in the sensory processing CMAC can be chosen so as to define the size of the neighborhood of generalization on the input space. This means that, as long as the regions of input space corresponding to pattern classes are reasonably well separated, the $G$ function can reliably distinguish one region of input space from another and hence classify the corresponding sensory patterns correctly.

In the case where the $D$ vector is time dependent, an extended portion of a trajectory $T_D$ may map into a single name $Q$ as shown in figure 21. It then is possible by integrating $Q$ over time and thresholding the integral to detect, or recognize, a temporal pattern $T_D$ such as a sound or a visual movement.

Note that the recognition, or naming, of a temporal pattern (as illustrated in figure 21) is the inverse of the decomposition of a task as illustrated in figures 14 thru 17 in the previous article in this series. In task decomposition a slowly varying command $C$ is decomposed into a rapidly changing output $P$. In pattern recognition a rapidly changing sensory experience $E$ is recognized by a slowly varying name $Q$. 

Figure 20. The $D$ vector is composed of sensory variables $E$ and context variables $R$. The function $G$ recognizes the existence of a $D$ vector in a particular region of pattern plus context space by outputting a $Q$ vector which is the name of that region.
Wordsmith is the video text editing system you've been waiting for. Its power, flexibility and simplicity help you carve any text editing task down to size—in a way you can understand. We wanted a system that allows you to think in traditional ways about text layout, yet at the same time makes the traditionally tedious operations such as cut and paste simple and fast. We think we've done it. We want you to decide for yourself.

Flexibility

- **Logical/Physical Page Distinction.** Define your own hardcopy size. Wordsmith remembers the difference between the screen size and the hardcopy page size.
- **Modular Hardcopy Driver.** Drive a Qume® Sprint-5 or TTY-like device directly now, Diablo, NEC and other hardcopy devices soon.
- **Pure Text.** Wordsmith files are pure text with no control characters mixed in. This universal format keeps you as compatible with the world as possible. What you see on the screen is what you get as hardcopy.
- **Page Templates.** Snapshots of the block layout of a page can be saved as named disk files, then later recalled and superimposed on the current page. Use such "templates" for standard multicolumn layouts, common letter formats, and fixed-field forms. A single keystroke dispatches you quickly from block to block as you fill in your page.

File Switching. Moving from document to document to examine, copy, move and change text is like rolling off a log. You're not confined to one disk file at a time anymore.

Simplicity

- **Auto Word Break.** Forget the right margin. Wordsmith notices when you won't be able to complete the current word and moves it to the next line for you as you continue typing.
- **Understandable Commands.** The most frequently used commands are single keystrokes. The rest are easily remembered abbreviations.
- **Informative Status Lines.** The top two screen lines constantly display page number information, document name, cursor position, tab stops and status/error phrases. You're always in touch with your document.

Protection Against Catastrophic Errors. It's nearly impossible to ruin your document with a single bad command. Wordsmith's page oriented design and double-checking user interface help you do what you mean!

The **Wordsmith**

TEXT EDITOR

Defining the New Generation of Text Editing

from Micro Diversions, Inc.

8455-D Tyco Rd.

Vienna, Va. 22180

(703) 827-0888

- Direct CP/M® and North Star DOS compatibility
- Available for 40x86, 24x80 and 16x64 memory-mapped video boards
- Fully reentrant for efficient multi-programming environments (8K program space, 5K data area)
- 8080 and Z80 compatibility

Ordering Information:

$200

(Screensplitter™ Owners $80)

Manual only $15

Check, VISA, Mastercharge

1. CP/M or North Star DOS version?
2. TTY or GUME interface?
3. Brand and memory address of video display board?
4. Ship on single or double density, 5" or 8" diskette?

Inquire about our custom keyboard.
Figure 21. A time varying $D$ vector traces out a trajectory $T_D$ which represents a sensory experience $T_E$ taking place in the context $T_R$. A section of a $T_D$ trajectory which maps into a small region of $Q$ space corresponds to the recognition of an extended temporal pattern as a single event.

The Use of Context

It frequently occurs in pattern recognition or signal detection that the instantaneous value of the sensory input vector $E$ is ambiguous or misleading. This is particularly true in noisy environments or in situations where data dropouts are likely to occur. In such cases the ambiguity can often be resolved or the missing data filled in if the context can be taken into account, or if the classification decision can make use of some additional knowledge or well-founded prediction regarding what patterns are expected.

In CMAC the addition of context or prediction variables $R$ to the sensory input $E$ such that $D = E + R$ increases the dimensionality of the pattern input space. The context variables thus can shift the total input (pattern) vector $D$ to different parts of input space depending on the context. Thus, as shown in figure 22, the ambiguous patterns $E_1$ and $E_2$, which are too similar to be reliably recognized as being in separate classes, can easily be distinguished when accompanied by context $R_1$ and $R_2$.

In the brain, many variables can serve as context variables. In fact, any fiber carrying information about anything occurring simultaneously with the input pattern can be regarded as context. Thus context can be data from other sensory modalities as well as information regarding what is happening in the behavior-generating hierarchy. In many cases, data from this latter source is particularly relevant to the pattern recognition task, because the sensory input at any instant of time depends heavily upon what action is currently being executed. For example, information from the behavior-generating hierarchy provides contextual information necessary for the visual processing hierarchy to distinguish between motion of the eyes and motion of the room about the eyes.

In a classic experiment, von Holst and Mittelstaedt demonstrated that this kind of contextual data pathway actually exists in insects. They observed that a fly placed in a chamber with rotating walls will tend to turn in the direction of rotation so as to null the visual motion. They then rotated the fly's head 180° around its body axis (a procedure which for some reason is not fatal to the fly) and observed that the fly now circled endlessly. By attempting to null the visual motion it was now actually increasing it.

Later experiments with motion perception in humans showed that the perception of a stationary environment despite motion of the retinal image caused by moving the eyes is dependent on contextual information derived from the behavior-generating hierarchy. The fact that the context is actually derived from the behavior-generating hierarchy rather than from sensory feedback can be demonstrated by anesthetizing the eye muscles and observing that the effect depends on the intent to move the eyes, and not the physical act of movement. The perceptual correction occurs even when the eye muscles are paralyzed so that no motion actually results from the conscious intent to move.

CMAC as a Predictive Memory

Contextual information can also provide predictions of what sensory data to expect. This allows the sensory processing modules
to do predictive filtering, to compare incoming data with predicted data, and to "fly-wheel" through noisy data or data dropouts.

The mechanism by which such predictions, or expectations, can be generated is illustrated in figure 23. Here contextual input for the sensory processing hierarchy is shown as being processed through a CMAC M module before being presented to the sensory pattern recognition G modules at each level. Inputs to the M modules derive from the P vector of the corresponding behavior-generating hierarchy at the same level, as well as an X vector which includes context derived from other areas of the brain, such as other sensory modalities or other behavior-generating hierarchies. These M modules compute \( R = M(P + X) \). Their position in the links from the behavior-generating to the sensory processing hierarchies allows them to function as a predictive memory.

They are in a position to store and recall (or remember) sensory experiences (E vector trajectories) which occur simultaneously with P and X vector trajectories in the behavior-generating hierarchy and other locations within the brain. For example, data may be stored in each M module by setting the desired output \( R_i \) equal to the sensory experience vector \( E_i \). At each instant of time \( t = k \), sensory data represented by \( E_k^i \) will then be stored on the set of weights selected by the \( P_k + X_k \) vector. The result will be that the sensory experience represented by the sensory data trajectory \( T_{P_i} \) will be stored in association with the context trajectory \( T_{P_i + X_i} \).

Any time afterwards, \( t = k + j \), a reoccurrence of the same context vector \( P_{k+j} + X_{k+j} \) will produce an output \( R_{k+j} \) equal to the \( E_{k} \) stored at time \( t = k \). Thus a reoccurrence of the same context trajectory \( T_{P_i + X_i} \) will produce a recall trajectory \( T_{R_i} \) equal to the earlier sensory experience represented by the sensory experience vector \( E_{R_i} \).
experience $T_{E_i}$. These predictive memory modules thus provide the sensory processing hierarchy with a memory trace of what sensory data occurred on previous occasions when the motor generating hierarchy (and other parts of the brain) were in similar states along similar trajectories. This provides the sensory processing system with a prediction of what sensory data to expect. What is expected is whatever was experienced during similar activities in the past.

In the ideal case, the predictive memory modules $M_i$ will generate an expected sensory data stream $T_{R_i}$ which exactly duplicates the observed sensory data stream $T_{E_i}$. To the extent that this occurs in practice it enables the $G_i$ modules to apply very powerful mathematical techniques to the sensory data. For example, the $G_i$ modules can use the expected data $T_{R_i}$ to:

- Perform cross-correlation or convolution algorithms to detect sync patterns and information bearing sequences buried in noise.
- Flywheel through data dropouts and noise bursts.
- Detect (or recognize) deviations or even omissions from an expected pattern as well as the occurrence of the pattern in its expected form.

If we assume, as shown in figure 23, that predictive recall modules exist at all levels of the processing-generating hierarchy, then it is clear that the memory trace itself is multi-leveled. In order to recall an experience precisely at all levels, it is necessary to generate the same context (ie: $P_i + X_i$ address) at all levels as existed when the experience was recorded.

**Internal World Model**

We can say that the predictive memory

---

**PET PRINTER ADAPTER**

GET HARD COPY FROM YOUR COMMODORE PET USING A STANDARD RS-232 PRINTER

The Cmc ADA 1200 drives an RS-232 printer from the PET IEEE-488 bus. Now, the PET owner can obtain hard copy listings and can type letters, manuscripts, mailing labels, tables of data, pictures, invoices, graphs, checks, needlepoint patterns, etc., using a standard RS-232 printer or terminal.

$98.50$ ADA 1200B
Assembled and tested

$169.00$ ADA 1200C
With case, power supply and RS-232 connector

Order direct or contact your local computer store. Add $3.00 for postage and handling per order.

CONNECTICUT microCOMPUTER
150 POCONO RD. BROOKFIELD, CT 06804
(203) 775-9659 TLX: 7104560052
modules $M_i$ define the brain's internal model of the external world. They provide answers to the question, "If I do this and that, what will happen?" The answer is that whatever happened before when this and that was done will probably happen again. In short, IF I do $Y$, THEN $Z$ will happen when $Z$ is whatever was stored in predictive memory the last time (or some statistical average over $N$ last times) that I did $Y$, and $Y$ is some action such as performing a task or pursuing a goal in a particular environment or situation, which is represented internally by the $P$ vectors at the various different levels of the behavior-generating hierarchy and the $X$ vectors describing the states of various other sensory processing behavior-generating hierarchies.

The $M_i$ modules (as all CMAC modules) can be thought of as storing knowledge in the form of IF/THEN rules. The CMAC property of generalization produces a recall vector $R_i$ (a THEN consequent) which is similar to the stored experience so long as the context vector $P_i + X_i$ (the IF premise) is within some neighborhood of the context vector during storage.

Much of the best and most exciting work now going on in the field of artificial intelligence revolves around IF/THEN production rules, and how to represent knowledge in large computer programs based on production rules. Practically any kind of knowledge, or set of beliefs, or rules of behavior can be represented as a set of production rules. The CMAC hierarchy shown in figure 23 illustrates how such computational mechanisms can arise in the neurological structure of the brain.

Conclusion

We have now completed the second step in our development. I have described a neurological model which can store and recall (and hence compute) a broad class of mathematical functions. I have shown how a hierarchical network of such models can execute tasks, seek goals, recognize patterns, remember experiences, and generate expectations. The final part of this series will include a brief overview of evidence that such networks actually exist in the brain. Also, this part will describe how a CMAC hierarchy can create plans, solve problems, and produce language. Finally I will discuss the design of robot control systems incorporating these properties and offer some suggestions as to how brain-like computing networks might be constructed and trained.

### ADDITIONAL READING

INTRODUCING BMT DISK SYSTEM AVAILABLE FOR
FEATURES:

- LANGUAGE TRANSLATION &

I DON'T,
PARLÉZ

- OVER 4,500 WORDS
- 60 PHRASES
- SCREEN OR PRINTER OUTPUT
- ALPHABETICAL LISTING
- AND MUCH MORE

P&H Linguistics, Inc.
56 Bay State Rd, Boston Mass, 02215
(617) 266 4688
MC & VISA ACCEPTED

PÄRLÈZ VOUS FRANÇAIS ?
I DON'T,

BUT MY TRS-80* does.

INTRODUCING TLF-50®, THE ONLY TURN-
KEY DISK SYSTEM AVAILABLE FOR
LANGUAGE TRANSLATION &

FEATURES:

(TRANSLATOR) (INSTRUCTOR)

- OVER 4,500 WORDS
- FRENCH GRAMMAR
- CONJUGATIONS
- VOCABULARY AND SPELLING DRILLS
- PRONUNCIATION
- METRIC CONVERSIONS

INTRODUCTORY OFFER $30.

July 2-13, Applications of Microcomputers to Life Science Education, Michigan Technological University, Houghton MI. The emphasis of this workshop will be on the use and development of educational computer models and simulations. The workshop will be set up to provide a maximum interaction with the microcomputer. Discussions will focus on ways of employing computer modeling techniques in undergraduate course work. Contact Dept of Biological Sciences, Michigan Technological University, Houghton MI 49931.

July 9-20, Computing Systems Reliability, University of California, Santa Cruz CA. Contact Institute in Computer Science, University of California Extension, Santa Cruz CA 95064.

July 11-13, Microcomputer Applications, Southern Technical Institute, Marietta GA. The emphasis of this seminar will be on the applications of microcomputers in industry. Software, hardware and interfacing techniques will be discussed. Contact Dr Richard L Castelluccius, Southern Technical Institute, Electrical Engineering Technology Dept, 534 Clay St, Marietta GA 30060.

July 16-18, Software Engineering, Crystal City Marriott, Arlington VA. This seminar is intended to familiarize the project manager, the system analyst, and the application programmer with techniques of developing software to meet user needs. Contact Information Technology Inc, POB 10129, Austin TX 78766.

July 16-18, Data Base Management, Crystal City Marriott, Arlington VA. This seminar is intended to familiarize the application programmer, data processing manager, and software system engineer with the latest techniques for the implementation and use of data base management. Contact Information Technology Inc, POB 10129, Austin TX 78766.

July 16-27, Introduction to Digital Electronics and Microcomputer Interfacing, Lexington VA. This hands-on laboratory course is for academic and industrial personnel. There will be approximately 60 hours of laboratory instruction with one microcomputer laboratory station for each two participants. Contact Prof Philip Peters, Dept of Physics, Virginia Military Institute, Lexington VA 24450.

July 19-20, Project Management, Crystal City Marriott, Arlington VA. The purpose of this seminar is to provide a basic understanding of the methodologies, tools, techniques and skills of software management. Contact Information Technology Inc, POB 10129, Austin TX 78766.

July 19-20, Structured Programming, Crystal City Marriott, Arlington VA. This course is aimed at both programmers and managers. It will cover an integrated set of software development techniques that can be scaled up for any size project development. It supports the development of error free programs by providing the programmer with effective means of controlling the design and code through continual validation checks. Contact Information Technology Inc, POB 10129, Austin TX 78766.

July 19-20, BASIC: A Computer Language for Executives, New York NY. Executive computing, problem solving, planning, forecasting and database systems will be discussed. Also to be covered are programming fundamentals, the mindless computer, sequence, decision and iteration, computer languages and BASIC. Contact American Management Associations, 135 W 50th St, New York NY 10020.

July 23-27 Finite Element Method in Mechanical Design, University of Michigan, Ann Arbor MI. This course is intended for engineers working in mechanical design where knowledge of stresses, displacements, or vibratory motion is important. No previous experience with finite elements is assumed. The course will familiarize the attendee with finite element modeling concepts and will review the fundamentals on which the method is based. Contact Engineering Summer Conferences, 400 Chrysler Cir, North Campus, University of Michigan, Ann Arbor MI 48109.
August 1-3, Microcomputer Applications, Southern Technical Institute, Marietta GA. The emphasis of this seminar will be on the applications of microcomputers in industry. Software, hardware and interfacing techniques will be discussed. Contact Dr. Richard L. Castellucci, Southern Technical Institute, Electrical Engineering Technology Dept, 534 Clay St, Marietta GA 30060.

August 6-8, Pattern Recognition and Image Processing, Hyatt Regency Chicago O'Hare, Chicago IL. This conference is sponsored by the Machine Intelligence and Pattern Analysis Committee of the IEEE Computer Society. The program will consist of submitted and invited papers and a large trade show of graphics and image processing equipment. Contact PRIP 79, POB 639, Silver Spring MD 20901.

August 6-10, SIGGRAPH '79, Chicago IL. This sixth annual conference on computer graphics will feature tutorials, technical sessions and an exposition of state-of-the-art computer graphics and image processing equipment. Contact Maxine D Brown, SIGGRAPH '79, Exhibition, Hewlett-Packard, 19400 Home- stead Rd, Cupertino CA 95014.

August 6-10, Modern Communication Systems: Analysis and Design, University of Southern California, Los Angeles CA. This course is devoted to the analysis and design of modern communication systems, with emphasis on the derivation of practical design equations useful for trade-off studies and overall synthesis. Contact University of Southern California, Continuing Engineering Education, Los Angeles CA 90007.

August 6-10, Advanced Microcomputer System Development: High Level Languages, Technology Trends, and Hands-On Experience, University of Southern California, Los Angeles CA. This course is intended to present the participants with a clear picture of the microcomputer revolution, provide hands-on programming experience using Extended BASIC and FORTRAN, analyze technology trends in the microcomputer field, and assess the impact of VH5I/VLSI. Contact University of Southern California, Continuing Engineering Education, Los Angeles CA 90007.

August 8-10, SIGPLAN Symposium on Compiler Construction, Boulder CO. This symposium will consider methods of, and experience with, constructing compilers. The emphasis will be less on theoretical methods, and more on techniques applied to real compilers. Contact Professor Leon Osterweil, Dept of Computer Science, University of Colorado, Boulder CO 80309.

August 8-10, First Annual Conference on Research and Development in Personal Computing, Hyatt Regency O'Hare, Chicago IL. This conference is sponsored by the Association for Computing Ma-

chinery (ACM) Special Interest Group on Personal Computing (SIGPC). A large trade show of personal computer and graphics equipment is planned to accompany an assortment of papers, panels, user group meetings, workshops, and person-to-person poster booths. Contact Bob Garofini, Computer Science Division, Dept of Mathematical Sciences, 300 Minard Hall, North Dakota State University, Fargo ND 58102.

August 13-15, Conference on Simulation, Measurement and Modeling of Computer Systems, Boulder CO. This conference will feature performance prediction techniques employed during the design, procurement and maintenance of computer systems. It will provide a forum for both applied and theoretical work in the disciplines of performance monitoring, modeling, and simulation of computer systems. Contact Gary Nutt, Xerox PARC, 3333 Coyote Hill Rd, Palo Alto CA 94304.

August 13-16, Q-GERT Network Modeling and Analysis, Ramada Inn, Lafayette IN 47905. This conference will provide the attendee with the information necessary to model complex systems using Q-GERT. Emphasis will be on the procedures for modeling and analysis. Contact Pritsker and Associates Inc, POB 2413, W Lafayette IN 47906.

August 13-17, High Speed Computation: Vector Processing, The University of Michigan, Ann Arbor MI. In this course, the architectural, software, and algorithmic issues of vector architecture are coordinated by the discussion of concepts in computer architecture, and by detailed study of current vector processors and their use. Contact Engineering Summer Conferences, 400 Chrysler Ctr, North Campus, The University of Michigan, Ann Arbor MI 48109.

August 19-22, International Conference on Computing in the Humanities, Dartmouth College, Hanover NH. This conference is intended to foster computer research and technique in all areas of humanistic study; to promote international cooperation in the development of programs, data banks, and equipment; and to make the results of research available. The program will include a plenary session each evening and shorter sessions during the day. Contact Stephen V F Waite, Kiewit Computer Ctr, Dartmouth College, Hanover NH 03755.

August 19-24, 1979 Symposium for Innovation in Measurement Science, Hobart and William Smith Colleges, Geneva NY. Sponsored by the Scientific Instrumentation and Research Division of the Instrument Society of America, scheduled sessions at this symposium...
include innovation in computers and electronics, mass flow measurement, chemical analysis, applied analysis in instrument control, physical analysis, medical instrumentation, and advances in Industrial measurement. Contact Instrument Society of America, 400 Stanwick St, Pittsburgh, PA 15222.

August 23-26, National Small Computer Show, New York Coliseum, New York NY. Exhibitions will include those of major manufacturers, distributors, and publications in the small computer field. A lecture series will include topics of interest to business and professional people, hobbyists and the general public. Contact National Small Computer Show, 74 E 56th St, New York NY 10022.

September 4-6, International Conference and Exhibition on Engineering Software, University of Southampton, England. The aim of this conference is to provide a forum for the presentation and discussion of recent advances in engineering software and the state-of-the-art in this field. The exhibition, held in conjunction with the conference, will cover all software products, services and equipment related to engineering software. Contact Dr R Adey, Engsoft, 6 Cranbury Place, Southampton SO2 OLG, ENGLAND.

September 4-7, Compon Fall '79, Capital Hilton Hotel, Washington DC. This 18th IEEE Computer Society International conference will present the latest developments in microprocessor architecture, support software, operating systems, and peripheral devices. Contact IEEE Computer Society, POB 639, Silver Spring MD 20901.

September 5-8, Info/Asia, Ryutsu Center, Tokyo. This exhibition will be devoted to information management, computers, word processing and advanced business equipment. The exhibition will be accompanied by a four day conference. Contact Clapp and Poliak Inc., 245 Park Av, New York NY 10017.

September 18-20, Wescon/79, St Francis Hotel, San Francisco CA, Contact Electronic Conventions Inc, 999 N Sepulveda Blvd, El Segundo CA 90245.

September 25-27, WPOE '79, San Jose Convention Ctr, San Jose CA. This show will provide a dedicated word processing and office/business equipment, services and materials. Complementing the exhibit will be a three day executive conference program that focuses on emerging technologies and their applications in the office. Contact Cardilige and Associates Inc., 491 Macara Ave, Suite 1014, Sunnyvale CA 94086.

September 26-29, MIMI '79, Queen Elizabeth Hotel, Montreal, Canada. This symposium is intended as a forum for the presentation and discussion of recent advances in mini and microcomputers and their applications. Special emphasis will be given to the theme of the conference, “The Evolving Role of Minis and Micros Within Distributed Processing.” Contact The Secretary, MIMI '79 Montreal, POB 2481, Anaheim CA 92804.

September 28-30, Northeast Personal and Business Computer Show, Hynes Auditorium, Boston MA. Displays and exhibits will showcase microcomputers and small computer systems of interest to businesspeople, hobbyists, professionals, etc. Lectures and seminars will be presented for all categories and levels of enthusiasts, including introductory classes for novices. Contact Northeast Exposition, POB 678, Brookline MA 02147.*

**Letters**

Text continued from page 6: written) would only perform garbage collection.

There are more facilities which could be added to this simple data base structure, but it would probably be better to stop at this point.

Jack L Warner
Bell Laboratories
600 Mountain Ave
Murray Hill NJ 07974

HAMMING ERROR CORRECTING CODE HAS PROBLEMS?

Michael Wimble recently described a method for storing coded data using a Hamming error correcting code which will correct a single bit error and detect double bit errors (February 1979 BYTE, page 180). It is very similar to a scheme I have used successfully for several years and recently published (Computer Design, September 1978).

Mr. Wimble's scheme, however, will cause havoc with some data recording devices as some of his coded bytes are exactly the same as some of the common control characters, i.e.: his data 3, coded as a hexadecimal 93, is identical to the ASCII Device Control 3 character, and data E, coded as a hexadecimal 4E, is the ASCII Record Separator. The latter should not cause trouble, but the former will automatically activate or deactivate some kinds of papertape terminals.

If this is likely to be a problem on the equipment you are using, one simple way to overcome it is to omit the P4 parity bit altogether and use strictly 7 bit codes. The media channel on which the omitted bit would have been stored is then placed so that no objectionable control characters are ever generated (except, of course, by error). Programs to generate and decode such schemes are presented in the Computer Design paper cited above.

You are not able to detect double errors using only 7 bits, but if your equipment is that bad it's probably time to pack it in anyway.

George White
Institut de Recherche d'Informatique et d'Automatique
Domaine de Voluceau
Roquencourt BP 105
78150 Le Chesnay FRANCE

**COMMENTS FROM A CHESS MASTER**

I was most flattered to read the story about my chess match with CHESS 4.7 (“Chess 4.7 versus David Levy” December 1978 BYTE, page 84) and I am delighted that you have been giving such excellent coverage in the pages of your magazine.

Since your article appeared I have been plagued by people writing to ask whether I have collected the $2,500 that I won in the bet. Professors Donald Michie (Edinburgh University), John McCarthy (Stanford University) and Seymour Papert (MIT) paid promptly and with great sportsmanship, just as I would have done had I lost the bet. Edward Kozdrowicki (Aerospace Corporation, El Segundo CA) has refused all attempts to persuade him to pay.

I hope that this will answer any further readers who might be curious about the bet.

David N. L. Levy
104 Hamilton Terrace
London NW8 9UP
ENGLAND *

**BYTE's Bugs**

Community Bulletin Board Correction

In the BYTE News for April 1979 (page 195) we mentioned that there was a PCNET run by the Chicago Area Computer Hobbyist Exchange. We should have said that a Community Bulletin Board is privately run by Ward Christensen and Randy Suess.

Correction

In May’s “What’s New” on page 254 we listed Semionics Associates' REM S-100 board as having a capacity of 8 K bytes and priced at $525. This should have read “The REM S-100 adds-in recognition memory board has a capacity of 4 K bytes and is priced at $345.”

**Trap Door Trap**

For shame! The National Bureau of Standards standard data encryption algorithm is not a trap-door algorithm. That term refers to one-way or public key systems.

Don McClains
Computer Systems Consultant
41 Washburn Pk
Rochester NY 14620

Oops! You, along with several other people, caught us with that one. [RGAC]
FCC TRYING TO CRACK DOWN ON TV INTERFERENCE. The Federal Communications Commission (FCC) has asked Atari, Apple, Commodore, Heath, Southwest Technical Products, and Radio Shack to submit their personal computer systems for TV interference testing. The systems made by these companies are presently exempted from FCC regulations since they are not directly connected to a TV set. However, there have been complaints regarding radio frequency (RF) interference from personal computer systems, and the FCC has decided to develop regulations regarding permissible RF radiation levels.

The computer manufacturers involved have indicated a willingness to cooperate with the FCC’s effort. The regulations could fine noncomplying manufacturers and permit the issuance of cease and desist orders. Some industry experts feel that a few manufacturers’ computer systems would not pass the FCC regulations.

INTEL ENHANCES 8086 FAMILY WITH I/O PROCESSOR. Intel continues to lead the way in microprocessor and microcomputer systems. Recently they announced the 8089, an I/O (input/output) processor to work with the 8086 16 bit microprocessor. This processor can more than double the performance of the 8086 by relieving it of I/O operations, much like the communications channel on an IBM 370.

$200 DISK SYSTEM EXPECTED BY YEAR END. Shugart and Matsushita Electric of Japan have signed an agreement whereby Matsushita will manufacture a low cost version of Shugart’s popular minifloppy disk drive. The drive is expected to sell for $50 in large OEM quantities and retail at about $125. Add to this the interface/controller circuitry, and the total retail cost should work out to a little over $200. This is less than a third of the price of current minifloppy systems. Matsushita expects to be making 100 drives per hour by year end.

The drive will store 70 K bytes, use a new head design, and be housed in sheet metal rather than cast aluminum. It will be only 2 inches high, half the height of the current drive. An industrial version with heavy duty components will be sold at $65 OEM.

Nippon Electric (NEC) is also rumored to be developing a low cost 5 inch disk drive.

14 MILLION MICROPROCESSORS SOLD LAST YEAR. That’s right, 14 million microprocessors were manufactured in 1978. One million 8 bit microprocessors and 13 million 4 bit microprocessors were made. If you didn’t realize it already, most were used in games. The most manufactured microprocessors were the 8 bit 6502 and the 4 bit TMS-1000. However, sales of electronic games using microprocessors have recently taken a sharp drop. Hence, the probability exists that there may be a slight decrease in microprocessor production in 1979.

16 BIT MICROPROCESSOR PICTURE STILL FUZZY. It is beginning to look as if Intel may have taken the right approach with the 8086 by designing a part which could be placed in production far ahead of the Zilog Z-8000 or Motorola 68000. They have over a year’s head start compared to the Z-8000 and possibly another half year’s lead over the 68000.

The 8086 part is far simpler than the Z-8000 or 68000 parts, and as a result it is closer to the earlier generations of microprocessors. Support parts for the 8086 such as the new 8087 floating point coprocessor also give the 8086 a commanding availability lead over the other two contenders at this time. All three machines are aimed at the high end of microcomputer application, providing significant computational power equivalent to traditional mini and main frame computers.
In the meantime, the traditional minicomputer manufacturers are not sitting still. Digital Equipment Corp (DEC), the largest of the "old-time" minimakers, has created an integrated circuit manufacturing facility to make its own 16 bit microprocessor (the LSI-11). This fall they will be making a super-micro mainframe, called the LSI-11/23, that will have almost all the power of a standard minicomputer at a fraction of the price ($6,800 compared to $12,000).

THE JAPANESE ARE COMING. Although last year a few Japanese electronics manufacturers introduced personal computer systems on their own home ground, none, so far, has ventured into the US market. This is probably due to the competition that already exists and the lack of sufficient price markup on personal computer systems. If the Japanese enter the personal computer market, it will probably be in the peripherals area.

However, the first major Japanese manufacturer has entered the small business computer market. NEC has introduced its ASTRA series of 16 bit microcomputer systems that start at $13,000 and range up to $130,000. The video terminal employs a Z-80 processor.

NEW BUBBLE MEMORY TECHNOLOGY. In a paper delivered by a Bell Laboratory researcher at a recent conference, it was disclosed that Bell Labs has made a major breakthrough in bubble memory technology. This breakthrough will mean a four times increase in storage size, a substantial decrease in cost and ten times faster operating speed. Although Texas Instruments and Rockwell have been in production on bubble memory devices for nearly a year, their high cost and small storage capability have prohibited their wide use. This new development, which will still take a few years to reach the market, should have a large impact on the mass storage area, particularly floppy disks.

The new device replaces the drive coils used in present bubble memories with wafer-thin conductive layers of gold or aluminum overlaid on the garnet structure. A current flows through these layers forming tiny magnetic fields around holes etched into the surface. The polarity of these fields controls the bubble movements.

By eliminating the costly and bulky coil structure, a new pathway design became available which provides a fourfold increase in storage capacity, is easier and less costly to produce, and reduces integrated circuit size, thereby reducing travel time.

IBM also announced that it has fabricated bubble memory devices that are 1 square inch in size and contain 25 M bits. These devices were made at the IBM Research Center in Yorktown Heights NY.

In the meantime, TI and Rockwell are currently sending out samples of their 256 K bit bubble devices and expect to be in production on these units by the end of the year. They expect to be sampling 1 M bit devices by the end of 1980, with production beginning in 1981.

On the whole, it does not appear that bubbles will provide any meaningful competition to floppies until the mid 1980s.

PERSONAL COMPUTER/CABLE TELEVISION SYSTEM PLANNED. Six Star Cablevision, a Los Angeles cable television outfit, will soon begin test marketing a personal computer system designed for use with a closed circuit TV system. Six Star will allocate 3 of 42 available channels to transmit data from data banks to subscribers. They claim to have 50 applications programs already prepared, which would be regenerated every 7 seconds. They plan to use a Mattel personal computer system with a printer, and charge $4 to $6 above the regular $7.50 monthly fee.
More BYTE BOOKS in your future...
THE BYTE BOOK OF COMPUTER MUSIC combines the best computer music articles from past issues of BYTE Magazine with exciting new material—all written for the computer experimenter interested in this fascinating field.

You will enjoy Hal Chamberlin's "A Sampling of Techniques for Computer Performance of Music", which shows how you can create four-part melodies on your computer. For the budget minded, "A $19 Music Interface" contains practical tutorial information—and organ fans will enjoy reading "Electronic Organ Chips For Use in Computer Music Synthesis".

New material includes "Polyphony Made Easy" and "A Terrain Reader". The first describes a handy circuit that allows you to enter more than one note at a time into your computer from a musical keyboard. The "Terrain Reader" is a remarkable program that creates random music based on terrain maps.

Other articles range from flights of fancy about the reproductive systems of pianos to Fast Fourier transform programs written in BASIC and 6800 machine language, multi-computer music systems, Walsh Functions, and much more.

For the first time, material difficult to obtain has been collected into one convenient, easy to read book. An ardent do-it-yourselfer or armchair musicologist will find this book to be a useful addition to the library.

Editor: Christopher P. Morgan
Pages: approx. 128
Price: $10.00

SUPERWUMPUS is an exciting computer game incorporating the original structure of the WUMPUS game along with added features to make it even more fascinating. The original game was described in the book What To Do After You Hit Return, published by the People's Computer Company. Programmed in both 6800 assembly language and BASIC, SUPERWUMPUS is not only addictively fun, but also provides a splendid tutorial on setting up unusual data structures (the tunnel and cave system of SUPERWUMPUS forms a dodecahedron). This is a PAPERBYTE™ book.

ISBN 0-931718-03-1
Author: Jack Emmerichs
Pages: 96
Price: $6.00

TINY ASSEMBLER 6800, Version 3.1 is an enhancement of Jack Emmerichs' successful Tiny Assembler. The original version (3.0) was described first in the April and May 1977 issues of BYTE magazine, and later in the PAPERBYTE™ book TINY ASSEMBLER 6800 Version 3.0.

In September 1977, BYTE magazine published an article entitled, "Expanding The Tiny Assembler". This provided a detailed description of the enhancements incorporated into Version 3.1, such as the addition of a "begin" statement, a "virtual symbol table", and a larger subset of the Motorola 6800 assembly language.

All the above articles, plus an updated version of the user's guide, the source, object and PAPERBYTE™ bar code formats of both Version 3.0 and 3.1 make this book the most complete documentation possible for Jack Emmerichs' Tiny Assembler.

ISBN 0-931718-08-2
Author: Jack Emmerichs
Pages: 80
Price: $9.00

A walk through this book brings you into Ciarcia's Circuit Cellar for a detailed look at the marvelous projects which let you do useful things with your microcomputer. A collection of more than a year's worth of the popular series in BYTE magazine, Ciarcia's Circuit Cellar includes the six winners of BYTE's On-going Monitor Box (BOMB) award, voted by the readers themselves as the best articles of the month: Control the World (September 1977), Memory Mapped IO (November 1977), Program Your Next ERROM in BASIC (March 1978), Tune In and Turn On (April 1978), Talk To Me (June 1978), and Let Your Fingers Do the Talking (August 1978).

Each article is a complete tutorial giving all the details needed to construct each project. Using amusing anecdotes to introduce the articles and an easy-going style, Steve presents each project so that even a neophyte need not be afraid to try it.
BASEX, a new compact, compiled language for microcomputers, has many of the best features of BASIC and the 8080 assembly language—and it can be run on any of the 8080 style microprocessors: 8080, Z-80, or 8085. This is a PAPERBYTE™ book.

Subroutines in the BASEX operating system typically execute programs up to five times faster than equivalent programs in a BASIC interpreter—while requiring about half the memory space. In addition, BASEX has most of the powerful features of good BASIC interpreters including array variables, text strings, arithmetic operations on signed 16 bit integers, and versatile IO communication functions. And since the two languages, BASEX and BASIC, are so similar, it is possible to easily translate programs using integer arithmetic data from BASIC into BASEX.

The author, Paul Warme, has also included a BASEX Loader program which is capable of relocating programs anywhere in memory.

SIMULATION is the second volume in the Programming Techniques series. The chapters deal with various aspects of specific types of simulation. Both theoretical and practical applications are included. Particularly stressed is simulation of motion, including wave motion and flying objects. The realm of artificial intelligence is explored, along with simulating robot motion with the microcomputer. Finally, tips on how to simulate electronic circuits on the computer are detailed.

RA6800ML: AN M6800 RELOCATABLE MACRO ASSEMBLER is a two pass assembler for the Motorola 6800 microprocessor. It is designed to run on a minimum system of 16 K bytes of memory, a system console (such as a Teletype terminal), a system monitor (such as Motorola MIKBUG read only memory program or the ICOM Floppy Disk Operating System), and some form of mass file storage (dual cassette recorders or a floppy disk).

The Assembler can produce a program listing, a sorted Symbol Table listing and relocatable object code. The object code is loaded and linked with other assembled modules using the Linking Loader LINK68. (Refer to PAPERBYTE™ publication LINK68: AN M6800 LINKING LOADER for details.)

There is a complete description of the 6800 Assembly language and its components, including outlines of the instruction and address formats, pseudo instructions and macro facilities. Each major routine of the Assembler is described in detail, complete with flow charts and a cross reference showing all calling and called-by routines, pointers, flags, and temporary variables.

In addition, details on interfacing and using the Assembler, error messages generated by the Assembler, the Assembler and sample IO driver source code listings, and PAPERBYTE™ bar code representation of the Assembler's relocatable object file are all included.

This book provides the necessary background for coding programs in the 6800 assembly language, and for understanding the innermost operations of the Assembler.
LINK68: AN M6800 LINKING LOADER is a one pass linking loader which allows separately translated relocatable object modules to be loaded and linked together to form a single executable load module, and to relocate modules in memory. It produces a load map and a load module in Motorola MIKBUG loader format. The Linking Loader requires 2 K bytes of memory, a system console (such as a Teletype terminal), a system monitor (for instance, Motorola MIKBUG read only memory program or the ICOM Floppy Disk Operating System), and some form of mass file storage (dual cassette recorders or a floppy disk).

It was the express purpose of the authors of this book to provide everything necessary for the user to easily learn about the system. In addition to the source code and PAPERBYTE™ bar code listings, there is a detailed description of the major routines of the Linking Loader, including flow charts. While implementing the system, the user has an opportunity to learn about the nature of linking loader design as well as simply acquiring a useful software tool.

ISBN 0-931718-09-0
Authors: Robert D. Grappel & Jack E. Hemenway
Pages: 72
Price: $8.00
Winter 1979

TRACER: A 6800 DEBUGGING PROGRAM is for the programmer looking for good debugging software. TRACER features single step execution using dynamic break points, register examination and modification, and memory examination and modification. This book includes a reprint of "Jack and the Machine Debug" (from the December 1977 issue of BYTE magazine). TRACER program notes, complete assembly and source listing in 6800 assembly language, object program listing, and machine readable PAPERBYTE™ bar codes of the object code.

ISBN 0-931718-02-3
Authors: Robert D. Grappel & Jack E. Hemenway
Pages: 24
Price: $6.00

MONDEB: AN ADVANCED M6800 MONITOR-DEBUGGER has all the general features of Motorola's MIKBUG monitor as well as numerous other capabilities. Ease of use was a prime design consideration. The other goal was to achieve minimum memory requirements while retaining maximum versatility. The result is an extremely versatile program. The size of the entire MONDEB is less than 3 K.

Some of the command capabilities of MONDEB include displaying and setting the contents of registers, setting interrupts for debugging, testing a programmable memory range for bad memory locations, changing the display and input base of numbers, displaying the contents of memory, searching for a specified string, copying a range of bytes from one location in memory to another, and defining the location to which control will transfer upon receipt of an interrupt. This is a PAPERBYTE™ book.

ISBN 0-931718-06-6
Author: Don Peters
Pages: 88
Price: $5.00

BAR CODE LOADER. The purpose of this pamphlet is to present the decoding algorithm which was designed by Ken Budnick of Micro-Scan Associates at the request of BYTE Publications, Inc., for the PAPERBYTE™ bar code representation of executable code. The text of this pamphlet was written by Ken, and contains the general algorithm description in flow chart form plus detailed assemblies of program code for 6800, 6502 and 8080 processors. Individuals with computers based on these processors can use the software directly. Individuals with other processors can use the provided functional specifications and detail examples to create equivalent programs.

Author: Ken Budnick
Pages: 32
Price: $2.00

BYTE BOOKS Division • 70 Main Street • Peterborough, New Hampshire 03458

Please send the books I have checked.

☐ Computer Music $10.00
☐ SUPERWIMPUS $6.00
☐ Tiny Assembler (3.1) $9.00
☐ Circuit Cellar $8.00
☐ BASEX $8.00
☐ Program Design $6.00
☐ Check enclosed ☐ Bill Visa ☐ Bill Master Charge

Card No. ___________ Exp. Date ________

$ _____ Total Books
Add 60¢ per book

$ _____ Postage/Handling

$ _____ Grand Total

Name ____________________ Title ________ Company ____________

Street ____________________ City ___________ State/Province __________ Code __________
The Mathematics of Computer Art

Introduction

Computer scientists and personal computer enthusiasts have a great appreciation of the beauty and form of art. They often use the tools of their trade, the computer and its associated peripheral devices, to create works of art. These works express particular, somewhat algorithmic and mathematical tastes in art forms. Since the late 1960s the use of computers and computer controlled devices for the generation of this artwork (often in three dimensions) has been firmly established. (See references 3, 4, 8, 9, and 14.) A great portion of this artwork has relied heavily on the computer's ability to precisely manipulate numerical quantities to produce drawings or sculptures that express complex mathematical relationships. Drawings in this category include figures which show the relationships between the phase, amplitude, and periods of different trigonometric functions; graphs of functions of two or more variables; and moiré patterns that can express complex relationships by interaction between families of similar simple curves (see reference 13).

This is not to say that all or even the majority of computer art is inherently mathematical. Two of the latest crazes in computer art, the recreation of natural scenes and the randomly drawn picture (called "controlled serendipity" by one artist in reference 11), are in essence nonmathematical. This article, however, will be concerned only with those figures which have mathematics as their basis.

Among figures which rely heavily on mathematics, two classes can easily be separated. One class is distinguished by the fact that it is precisely the equations themselves which give the figures beauty and appeal. While even the mathematically uninitiated can perceive the beauty of these forms, only those who understand the underlying mathematics can fully appreciate the plots. Some examples of this class are the endless varieties of lissajous figures (see references 2 and 6), and two other famous trigonometrically based plots, "Sine Curve Man" (shown in figure 1; see reference 15) and "Christmas Wreath" (see reference 1). The beauty of "Sine Curve Man" is in part due to the undulating sine curves, differences in the phase

Figure 1: "Sine Curve Man" by Charles Csuri and James Shaffer, a trigonometrically based plot. Reprinted with permission from Computers and Automation, August 1967, Copyright 1967 and published by Berkeley Enterprises Inc, 815 Washington St, Newtonville MA 02160.

About the Author

Kurt Schmucker has been employed as a mathematician at the Department of Defense in Washington DC since 1974. He has masters degrees from both Michigan State University and Johns Hopkins University. He is now an advanced special student in the Computer Science Department at the University of Maryland and an assistant professorial lecturer in computer science at George Washington University. Mr Schmucker's current interests are in natural language processing and computer graphics.

Mr Schmucker is the author of "The Computers of Star Trek," which appeared in the December 1977 BYTE.
between the different curves, and the variation in the amplitudes. These form the mathematical base for the figure.

The other class of figures relies on mathematics not for the positioning of the actual lines but for the meaning or the importance of the resulting total plot. For these figures, the actual equations which are plotted are not as important as the relationships which are revealed. Some examples of this class are moiré patterns and projection plots of multi-dimensional figures (see reference 12).

An example of a moiré figure is shown in figure 2. Notice that the lines in this figure are nothing more than regularly spaced radii of two circles — lines whose equations are easily determined. What is fascinating is the complex interference pattern, a pattern which can express complex relationships between those lines. In this article, these two classes of figures will be discussed by examining in detail one example of each.

Crest

An example of a computer generated figure which relies on complex mathematical relationships for its beauty is the crest,

Figure 2: Moiré figure, an interference pattern between regularly spaced radii of two circles.
shown in figure 3 (see reference 5). While the beauty of this figure can be appreciated without examining its mathematics, a more complete understanding is necessary in order to reproduce it on a different computer or to fully comprehend the complexity of the figure. One can easily examine figure 3 and determine by its symmetry the decomposition which is shown in figure 4.

The basic unit of figure 3 is shown in figure 5. If the equations which generate the basic unit can be found, then the entire figure can be generated by appropriately manipulating these equations. In an analysis of the unit in figure 5, one can see that the equation of the outer envelope of lines is the only portion of real importance. An examination of this curve brings to mind the spirals studied when one first encounters the use of polar coordinates. There are a number of different kinds of such spirals, most notably the spiral of Archimedes, the parabolic spiral, and the logarithmic spiral. By comparing the graphs of these spirals to figure 5 it can be seen that the logarithmic spiral closely approximates the desired curve. Recall that a logarithmic spiral (shown in figure 6) has an equation of the form \( r = ae^{-\theta/b} \), where \( a \) and \( b \) are positive real numbers. By a suitable choice of the constants \( a \) and \( b \), along with some transformations applied to the equations of two such spirals, we will be able to obtain the equation of the desired envelope.

To find the equation of this envelope, the graph of the logarithmic spiral must be rotated, translated, and reflected. The fact that the curve is usually expressed in polar form simplifies this task considerably. All three of these transformations can be expressed much more easily in that system than in Cartesian coordinates. Figure 7 shows the resulting graphs and their equations as the graph of the spiral is progressively reflected about the y axis, rotated clockwise by 60°, and translated.

Superimposing the graph of:
\[
\begin{align*}
x &= -ae^{-\theta/b} \cos(\theta - \pi/3) + a \cos \pi/3 \\
y &= ae^{-\theta/b} \sin(\theta - \pi/3) + a \sin \pi/3
\end{align*}
\]
upon the last portion of figure 7, the graph in figure 8 is obtained, which is precisely the desired envelope.

Text continued on page 110.

---

**A COMPLETELY REFURBISHED "SELECTRIC" ASCII TERMINAL FOR THE SMALL BUSINESSMAN OR SERIOUS HOBBYIST.**

The AJ 841 I/O terminal. Now available from dealers nationwide.

Demand for our AJ 841 I/O computer terminal has been great. And now it's getting even greater. So call your local computer shop dealer right away. Supply is limited! You may never have another opportunity like this one to buy your own professional terminal.

The AJ 841 features:
- Choice of serial RS 232 or parallel interface
- ASCII code
- 14.9 cps printout
- High quality Selectric printing
- Heavy-duty Selectric mechanism
- Off-line use as typewriter
- Documentation included
- 30-day warranty on parts and labor (details available on request)

Call toll-free now
For location of your nearest AJ dealer, call toll-free:
800/538-9721
California residents call 408/263-8520.

ANDERSON JACOBSON

Circle 9 on inquiry card.

July 1979 © BYTE Publications Inc 107
DIGITAL RESEARCH

- CP/M* FDISC — Disk Operating System complete with Text Editor, Assembler, Debugger, File Manager and system utilities. Available for wide variety of disk systems including North Star, Helios II, Microline, ICOM (all systems) and Altair. Supports computers such as Sorcerer, Horizon, Sol System III, Versatile, Altair 8800, COMPAL-30, DYNABYTE DB/B, and ICOM Attaché. Specify desired configuration. $149/$25

- MAC — 8080 Macro Assembler. Full Intel macro definitions. Pseudo Ops include RFC, IRP, REPT, TITLE, PAGE, and MACLIB. Z-80 library included. Produces Intel absolute hex output plus symbolic output for use by SED (see below). $160/$15

- SED — 8080 symbolic debugger. Full trace, count and breakpoint program testing system with backward trace and histogram utilities. When used with MAC, provides full symbolic display of memory labels and equated values. $85/$15

- TEX — Text formatter to create paginated, page-numbered and justified copy from source text files. Directable to disk or printer. $85/$15

- DESPOOL — Program to permit simultaneous printing of data from disk while user executes another program from the console. $30/$1

MICROSOFT

- Disk Extended BASIC — New version, ANSI compatible with long variable names. WHILE, WEND, changing, variable length file records. $390/$25

- FORTRAN-80 — ANSI '66 except for COMPLEX plus many extensions. Includes relocatable object compiler, linking loader, library manager. Also includes MACRO-80 (see below). $490/$25

- COBOL-80 — ANSI 74 PI-2020 compiler with relocatable object runtime package. Format same as FORTRAN-80 and MACRO-80 modules. Complete ISAM, interactive ACCEPT/DIPLAY, COPY, EXTEND. $825/$25

- MACRO-80 — 8080/2080 Macro Assembler. Intel and Zilog mnemonics supported. Relocatable object linkable output. Loader, Library Manager and Cross Reference List utilities included. $149/$16

- MACRO-80 plus FORTRAN subroutine library. Library includes ABS, SIGN, EXP, LOG, SORT, DSO, ATAN, DATAN, etc. etc. $219/$15

- EDIT-80 — Very fast random access text editor for text with or without line numbers. Global and intra-line commands supported. File compare utility included. $69/$15

XITAN (software requires Z-80 CPU)

- Disk BASIC — Fast powerful interactive interpreter. PRIVACY password security. Can dynamically open a large number of files simultaneously for random or sequential I/O. $159/$30

- Z-TEL — Text editing language. Expression evaluation iteration and conditional branching ability. Registers available for text and commands. Macro command strings can be saved on disk for re-use. $89/$20

- ASM Macro Assembler — Mnemonics per Intel with Z-80 extensions. Macro capabilities with absolute Intel hex or relocatable object modules. $369/$20

- LINKER — Link-edits and loads ASM modules. $69/$20

- Z-BUG debugger — Traces break-point tester. Supports decimal, octal and hex modes. Disassembles to ASM mnemonic set. Emulation technique permits full tracing and breakpoint support through ROM. $89/$20

SOFTWARE SYSTEMS


- STRUCTURED SYSTEMS GROUP

- General Ledger — Interactive and flexible system providing proof and report outputs. Customization of COA created interactively. Multiple branch accounting allows bookkeeping performed at data entry for proof, COA correction etc. Journal entries may be batched prior to posting. Closing procedure automatically backs up input files. All reports can be tailored as necessary. Requires CBASIC. $899/$20

- Accounts Receivable — Open item system with output for internal aged reports and customer-oriented statement and billing purposes. On-Line Enquiry permits information for Customer Service and Credit departments. Interface to General Ledger provided for both systems used. Requires CBASIC. $699/$20

- Accounts Payable — Provides aged statement of accounts by vendor with check writing for selected invoices. Can be used alone or with General Ledger and/or NAD. Requires CBASIC. $699/$20

- NAD Name and Address selection system — interactive mailing list creation and maintenance program with output as full reports with reference data or restricted information for mail labels. Transfer system for extraction and transfer of selected records to create new files. Requires CBASIC. $799/$20

- QSORT — Fast sort/merge program for files with fixed record length, variable field length information. Up to five ascending or descending keys. Full back-up of input files created. Parameter file created, optionally with interactive program which requires CBASIC. Parameters for each generated file. $799/$20

- Word Master Text Editor — In one mode has super-set of CP/M's ED commands including global searching and replacing, forward and backwards in line. In video mode, provides full screen editor for users with serial addressable cursor terminal. $150/$25
Software for most popular 8080/280 computer disk systems, including NORTH STAR, iCOM, SD SYSTEMS, DYNABYTE DB8/2, HELIOS, ALTair, TRS-80, 8" IBM and OHIO SCIENTIFIC formats.
By redoing this work in rectangular coordinates, we can see how much easier it is to manipulate these equations in the polar coordinate system. The reflected, rotated, and translated coordinates of a point \((X,Y)\) can be calculated with the matrix equation which is called \(a\) in table 1.

Substituting the specific values needed to repeat the previous work and multiplying the three 3 by 3 matrices together, we obtain equation \(b\) in table 1. This is the same result obtained earlier.

It is now a trivial matter to obtain the lines in figure 5 by drawing chords between points selected equiangularly along each of the two curves. One can extend this by similar modifications to the equation \(r = ae^{-\theta/b}\) to obtain the crest in figure 3. The constants \(a\) and \(b\) determine the size of the resulting plot and the curvature of each of the six "leaves" respectively.

The Dissected Square

The plot in figure 9 is not too difficult to understand at first glance (see reference 7). In essence it is a set of concentric squares with the area between the squares divided into smaller squares. Postponing the detailed discussion until later, the figure can be constructed in the following manner: given a square with a side of length \(X\), construct a concentric square with a smaller side of length \(Y\). The value of \(Y\) is determined by \(X\) in a manner to be explained later, but note that \(Y < X\). Extend the sides of the smaller square until they meet the edges of the square of side \(X\). The intermediate result is shown in figure 10. Divide the shaded regions into squares. (It will be shown that this is always possible when \(X\) and \(Y\) are chosen carefully.) At this point, consider the square of side \(Y\) to be the outer square and...
The Software Company

Technical Systems Consultants, Inc.

6800 & 8080 Text Processing
TSC's Text Editing System and Text Processing packages, available for both 6800 and 8080, can turn your micro into a very powerful word processing system. Two steps are involved. First you edit a file to say what you want without worrying about justification, pagination, etc. The TSC editor is a very powerful line and content oriented editor which may be used for this purpose.

The second step is to 'process' the edited file. The TSC text processor reads the edited file and outputs it to a printer with right justification, pagination, indented page numbering and titling, indenting, margin control, title centering, and much more. Macros may be written in the text processor commands which allow complex operations such as footnotes, form letters, or almost anything imaginable.

Price includes manual with printed source listing and object code on tape or disk.

6800 Text Editing System
SL68-24C w/cassette $30.45
SL68-24D w/mini FLEX™ disk $31.50
SL68-24F w/8" FLEX™ 1.0 disk $50.00

8080 Text Editing System
SL80-10P w/paper tape $37.50
SL80-10F w/8" CP/M™ disk $40.00
SL80-10D w/5" HS CP/M™ disk $40.00

6800 Text Processing System
SL68-29C w/cassette $38.95
SL68-29D w/mini FLEX™ disk $40.00
SL68-29F w/8" FLEX™ 1.0 disk $75.00
SL68-29F2 w/5" FLEX™ 2.0 disk $65.00

8080 Text Processing System
SL80-11P w/paper tape $41.00
SL80-11F w/8" CP/M™ disk $50.00
SL80-11D w/5" HS CP/M™ disk $50.00

FLEX™ Software
The FLEX™ disk operating system is rapidly becoming the standard of 6800 disk users. It's powerful capabilities and ease of use make it a natural choice for any application from the most minimal hobbyist system to the most complete industrial development system.

FLEX™ for SWTPC and SSB SWTPC mini floppy users can now upgrade to FLEX™ 2.0, the same version of FLEX™ as found on the SWTPC 8" disk system. Simply add 8K of memory at $400 and boot up from a FLEX™ 2.0 disk. Also available for Smoke Signal disk users in a mini floppy version (FLEX™ 2.0) and a full sized version (FLEX™ 1.0). Includes TSC editor and assembler on disk. All manuals also supplied (no source listings).

FX02-SWT FLEX™ 2.0 for 5" SWTPc $75.00
FX02-SSB FLEX™ 2.0 for 5" SSB $90.00
FX01-SSB FLEX™ 1.0 for 8" SSB $100.00

Sort/Merge Package
Sorts any type file according to parameters you supply. Files too large to fit in memory are automatically split up, individually sorted, and merged into one output file. Accepts fixed or variable length records, up to 20 ascending or descending keys, non-ASCII sequence sorts, and much more.
No source listing. Specify mini FLEX™, FLEX™ 1.0, or FLEX™ 2.0.
AP68-10 6800 Sort/Merge $75.00
Manual Only $15.00

FLEX™ Utilities
Complete set of 36 additional utilities ranging from memory dump to disk diagnostics. Includes printed manuals and source listings with object code AND source listing on disk.
UV1-6D w/mini FLEX™ disk $99.95
UV1-6F w/8" FLEX™ 1.0 disk $124.95
UV1-6F2 w/5" FLEX™ 2.0 disk $119.95

TSC BASIC for 6800
The fastest floating point BASIC interpreter for any 8 bit microprocessor. Features include six digit floating point math, full transcendental functions, unlimited string length, if/then/else construct, logical operators, and two-dimensional arrays including string arrays.

FLEX™ disk versions support random as well as sequential access files (the mini FLEX™ version has only sequential). Random files accessed by true record I/O or by virtual arrays. Disk version also allows TRACE, a renumber facility, ON ERROR GOTO, PTR for obtaining the address of a variable, CHAIN, and a very powerful method of printing to any type and number of output devices.

SAVE and LOAD commands produce FLEX™ compatible files which can be externally edited while a COMPILE command places an unreadable, compressed form of the source on disk which can only be executed by RUN. This feature permits the distribution of proprietary BASIC programs.

Cassette version runs in a minimum of 12K, disk version should have at least 16K. No source listing is included.
AP68-11C w/cassette $39.95
AP68-11D w/mini FLEX™ disk $49.95
AP68-11F w/8" FLEX™ 1.0 disk $59.95
AP68-11F2 w/5" FLEX™ 2.0 disk $54.95

All orders should include 3% for postage and handling (8% on foreign orders). Send 25¢ for a complete software catalog.

Technical Systems Consultants, Inc.
Box 2574 W. Lafayette, IN 47906

*CP/M™ is a registered trademark of Digital Research
FLEX™ is a trademark of Technical Systems Consultants, Inc.

Circle 363 on inquiry card.
Figure 6: Logarithmic spiral with equation of form $r = ae^{-\theta/b}$ using polar coordinates.

$x = ae^{-\theta/b} \cos(\theta)$
$y = ae^{-\theta/b} \sin(\theta)$

Figure 7: Graphs and equations of a logarithmic spiral as it is reflected about the y axis, rotated clockwise by 60°, and translated.

$x = -ae^{-\theta/b} \cos(\theta + \pi/3) + a \cos(\pi/3)$
$y = ae^{-\theta/b} \sin(\theta + \pi/3) + a \sin(\pi/3)$

$x = -ae^{-\theta/b} \cos(\theta + \pi/3)$
$y = ae^{-\theta/b} \sin(\theta + \pi/3)$

Figure 8: Superimposition of graph of: $x = -ae^{-\theta/b} \cos(\theta - \pi/3) + a \cos \pi/3$; and $y = ae^{-\theta/b} \sin(\theta - \pi/3) + a \sin \pi/3$; which yields the desired envelope shape.

begin again by choosing a suitable $Y'$ where $Y' < Y$. This process is terminated when $Y'$ assumes a certain specified value. What is significant about this plot, however, is the mathematics that it represents. This figure proves the following theorem:

$$\sum_{i=1}^{n} i^3 = \left( \sum_{i=1}^{n} i \right)^2$$

for all positive integers, $n$ (an offshoot of the theorem of Nicomachus) for the case $n = 26$.

To see this, it is easier to examine the associated figure for a smaller $n$ than 26, say $n = 6$ (see figure 11). If the smallest squares in the center of the figure are taken as unit squares, then the area of the large square can be calculated in two different ways. In the first way, the lengths of two sides can be multiplied. Since we are dealing with squares, any two sides can be used. The left side is of length $6(6+1)$ or in general $n(n+1)$, as can be seen by considering the shaded squares which lie along the left side. The length of the opposite side can be calculated by considering the shaded squares which extend diagonally from the center to the right side to obtain:

$$b = 2(6 + 5 + 4 + 3 + 2 + 1)$$

or in general:

$$b = 2 \sum_{i=1}^{n} i$$

Therefore the area of the square is:

$$ab = 6(6+1) \times 2(6+5+4+3+2+1)$$

or in general:

$$ab = n(n+1) \times 2 \sum_{i=1}^{n} i$$

$$\sum_{i=1}^{n} i = 4 \left( \sum_{i=1}^{n} i \right)^2$$

However, the area of the square can also be calculated by summing the areas of all the component squares. There are four squares of area 1, eight squares of area 4, twelve of area 9, etc. Therefore the area of the large square is:

$$4 \times 1^2 + 4 \times 2^2 + 4 \times 3^2 + 4 \times 4^2 + 4 \times 5^2 + 4 \times 6^2$$

or in general:

$$\sum_{i=1}^{n} i^3$$
By equating these computations of area, the desired theorem is obtained.

In drawing this figure, one need only choose an X of the form \(n(n + 1)\) for the side of the largest square, where \(n\) is an integer greater than 3. The sides of the inner squares

\[
\begin{align*}
\begin{pmatrix} x' \ y' \ 1 \end{pmatrix} &= \begin{pmatrix} x \ y \ 1 \end{pmatrix} \\
&= \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} x \ y \ 1 \end{pmatrix}
\end{align*}
\]

Table 1: Matrix equation \(a\) reflects, rotates, and translates coordinates of a point \((X, Y)\). Matrix equation \(b\) has substituted in it the specific values needed to repeat the earlier equations. We obtain the same result.
Figure 10: A square with sides of length $X$ has constructed within it a concentric square with sides of length $Y$. The sides of the smaller square are extended until they meet the edges of the square of side $X$. The shaded regions are next divided into squares.

Figure 11: Dissected square for $n = 6$. The left side is of length $6(6+1)$.

Figure 12: Two plotting procedures for a square with vertices ABCD. The smallest amount of pen motion occurs at left when plotting begins at point $A$ with consecutive drawing movements to $B$, $C$, $D$, and then back to $A$. In this method the length of nondrawing moves is 0.

At right is seen a nonoptimal plotting scheme. Starting at $A$, the pen draws to $B$, a nondrawing move is made to point $D$, the pen draws from $D$ to $C$ and then to $B$, a move is made to $A$, and then pen draws from $A$ to $D$.

are also numbers of this form, obtained by decrementing $n$ by 1 for each successive new square. When $Y = 2$, the last two lines are drawn, completing the figure. This choice of $X$ and $Y$ always allows the shaded areas of figure 10 to be decomposed into squares, as they are all rectangles with one side of length $n$ and the other of $n(n - 1)$. A rectangle with these proportions is dissectible into $(n - 1)$ squares of side $n$.

Plotting Considerations and Implementation

In developing the software to produce these drawings, the logic used to understand the generation of the figures was extended into the implementation of the code. Although this solution to the problem works, it turns out to be grossly inefficient in construction and plotting time.

These figures are best plotted on a high speed incremental plotter using ink rather than a ballpoint pen. The use of ink in plotting immediately causes a 50% reduction in plotting speed in order to avoid smears on the final plot. This and the high density of lines required to produce an aesthetically pleasing picture resulted in an average plot time of two hours per figure.

With these two considerations in mind, it became desirable to optimize the required plot time by minimizing pen movement. In the plotting of figures like those above, the total pen movement is comprised of the movement used to reposition the pen prior to the drawing of a new line (ie: when the pen tip is in the up position) and the actual drawing of the line (ie: when the pen tip is in the down position, that is, is in contact with the plotting surface and is drawing). While the total length of the "draws" (ie: when the pen tip is down and drawing) is fixed for any given figure, the length of the "moves" is variable. The total plot time can be diminished by minimizing these moves.

Consider the plotting of a square whose vertices are ABCD (see figure 12). Let us
WE HAVE WHAT YOU WANT... WHEN YOU WANT IT.

You want speed? Except for the Heath H8 board, every Econoram* runs trouble-free with 4 MHz systems. You want low cost? We use low power static parts exclusively, which means our maximum guaranteed current specs are often lower than the typical current specs of our competitors. You want long cost? Our volume purchasing and technical know-how keeps our products price-competitive with boards that do a whole lot less. You want it now? Not only do we stock what we advertise, but many computer stores both nationwide and overseas stock much or all of the CompuKit line.

You want a choice of configurations? For lowest price buy an "unkit", with sockets and bypass caps pre-soldered in place for easy assembly; specify assembled and tested boards when you can't wait to get up and running; and if you're dealing with critical applications—like accounting or medical applications—check into the boards qualified under our high-reliability Certified System Component program. These boards are burned in for 200 hours minimum and are immediately replaced in the event of failure within 1 year of invoice date.

You want reliable, cost-effective operation? That's how we got our reputation... ask any Econoram owner.

FEATURED BOARD OF THE MONTH:
ECONORAM IIA $149 UNKIT, $164 ASSM

The 8K X 8 Econoram II has been the workhorse of many an S-100 computer system... and now it's improved, with features like new 4K static chips, guaranteed 4 MHz operation, simpler layout, and low power (draws 8W maximum from the 8V bus—and that's a 4 MHz spec). We've retained all the popular features of the original, such as deselect switch for 4K operation, dual 4K block configuration with independent addressing, and switchable memory protect for each block. The oldest board on the block is now one of the newest, and we think you're really going to like it.

OTHER ECONORAM* FULLY STATIC MEMORY BOARDS:

<table>
<thead>
<tr>
<th>Name</th>
<th>Storage</th>
<th>Buss</th>
<th>Speed</th>
<th>Configuration</th>
<th>Unkit</th>
<th>Assm</th>
<th>CSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Econoram IV</td>
<td>16K X 8</td>
<td>S-100</td>
<td>4 MHz</td>
<td>1-16K block</td>
<td>$295</td>
<td>$329</td>
<td>$429</td>
</tr>
<tr>
<td>Econoram VI</td>
<td>12K X 8</td>
<td>H8</td>
<td>2 MHz</td>
<td>1-8K, 1-4K</td>
<td>$200</td>
<td>$270</td>
<td>n/a</td>
</tr>
<tr>
<td>Econoram VII</td>
<td>24K X 8</td>
<td>S-100</td>
<td>4 MHz</td>
<td>2-4K, 2-8K</td>
<td>$445</td>
<td>$485</td>
<td>$605</td>
</tr>
<tr>
<td>Econoram IX</td>
<td>32K X 8</td>
<td>Dig Grp</td>
<td>4 MHz</td>
<td>2-8K, 1-16K</td>
<td>$649</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>Econoram X</td>
<td>32K X 8</td>
<td>S-100</td>
<td>4 MHz</td>
<td>2-8K, 1-16K</td>
<td>$599</td>
<td>$649</td>
<td>$789</td>
</tr>
<tr>
<td>Econoram XI</td>
<td>32K X 8</td>
<td>SBC</td>
<td>4 MHz</td>
<td>2-8K, 1-16K</td>
<td>n/a</td>
<td>n/a</td>
<td>$1050</td>
</tr>
</tbody>
</table>

BANK SELECT MEMORIES (for Alpha Micro Systems, Marinchip, etc.)

<table>
<thead>
<tr>
<th>Name</th>
<th>Storage</th>
<th>Buss</th>
<th>Speed</th>
<th>Configuration</th>
<th>Unkit</th>
<th>Assm</th>
<th>CSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Econoram XII-16</td>
<td>16K X 8</td>
<td>S-100</td>
<td>4 MHz</td>
<td>2 indep. banks**</td>
<td>$369</td>
<td>$419</td>
<td>$519</td>
</tr>
<tr>
<td>Econoram XII-24</td>
<td>24K X 8</td>
<td>S-100</td>
<td>4 MHz</td>
<td>2 indep. banks**</td>
<td>$479</td>
<td>$539</td>
<td>$649</td>
</tr>
<tr>
<td>Econoram XIII</td>
<td>32K X 8</td>
<td>S-100</td>
<td>4 MHz</td>
<td>2 indep. banks**</td>
<td>$629</td>
<td>$699</td>
<td>$849</td>
</tr>
</tbody>
</table>

* Econoram is a trademark of Bill Godbout Electronics.

** Econoram XII-16 and -24 have 2 independent banks addressable on 8K boundaries; Econoram XIII has 2 independent banks addressable on 16K boundaries.

NEW! 2708 EROM BOARD KIT... at a special introductory price.

4 independently addressable 4K blocks, with selectable disable for each block. Built to Econoram standards (dipswitch addressing, top quality board, sockets wave-soldered in place), and includes dipswitch selectable jump start built right into the board. Includes all support chips and manual, but does not include EROMs. Special introductory price through August 1, 1979: $69.95. After that, the price goes up to the normal $85...
don’t say we didn’t warn you.

POPULAR ICs AT POPULAR PRICES

Low power 2102s for 2 MHz systems on special: 10/$9.90. 1791 MOS LSI dual density disc controller from Western Digi-10 slot unkit: $90. 18 slot unkit: $124. Each motherboard includes all edge connectors wave-soldered in place for easy assembly, integral active termination circuitry, extra wide power and ground traces, and much more.

NEW MULLEN COMPUTER PRODUCTS: H8 EXTENDER BOARD KIT $39

We’re happy to distribute a kit that really takes the popular H8 computer to another level. Includes jumper links in the power supply lines for insertion of fuses, Ammeters, current limiters, etc.

ACTIVE TERMINATOR KIT $34.50

As written up in the April '79 issue of Kilobaud Microcomputing. Our much imitated design plugs into any S-100 motherboard to reduce ringing, crosstalk, noise, and other buss-related problems.

S-100 MOTHERBOARDS

11 slot unkit: $90. 18 slot unkit: $124. Each motherboard includes all edge connectors wave-soldered in place for easy assembly, integral active termination circuitry, extra wide power and ground traces, and much more.

FREE FLYER: We'll be glad to tell you more than the space of this ad permits. Just send your name and address, we’ll take care of the rest. If you’re in a hurry, enclose 41¢ in stamps for 1st class delivery.

TERMS: Allow 5% shipping, excess refunded. Cal res add tax. VISA®/Mastercharge® call our 24 hour order desk at (415) 562-0638. COD OK with street address for UPS COD charge applies. Prices good through cover month of magazine. Orders under $15 add $1 handling.

Circle 150 on inquiry card.
assume that vertex A is the origin of the plot. Clearly, the smallest amount of pen movement possible is 4s, where s is the length of the side of the square. The value of 4s is obtained when the plotting begins at point A (i.e., the origin) with consecutive draws to B, C, D, and then a final draw to A. In this case the length of the moves is 0. A nonoptimal plotting scheme for this figure would be to start at A, and then draw to B, move to D and draw to C and then to B, move to A and then draw to D. The total pen movement for this scheme is 5s + \sqrt{2}s, where again s is the length of the square. It should be clear that there is no upper limit on the total pen movement, as the moves have no effect on the resulting plot and can be increased without bound.

Unfortunately, it is not always possible to find a plotting scheme in which the length of the pen moves is 0. A simple plot for which the best possible plotting scheme includes some moves is shown in figure 13. If s is the side of the square, the best possible plotting scheme has a total pen movement of 5s + 2\sqrt{2}s (see reference 10).

Of the two figures discussed in detail, the crest and the dissected square, only the crest can be drawn with zero moves. The plotting scheme which obtains this optimal solution is shown in figure 14. Using this strategy resulted in a substantial savings in total plotting time.

Unfortunately, no plotting scheme for the dissected square which has zero moves is possible. In fact, no scheme was found which significantly reduced the total plot time from that obtained by using the notions explained in detail above. It is felt that this is because all the plotting schemes we tried involved decomposing long line segments into a number of smaller such segments which were not drawn consecutively. With an on-line incremental plotter this requires the processor controlling the pen to issue a much larger number of plot commands. In a multiprocessing environment, any advantage gained in the total length of the moves was completely eliminated by the increased processing time with its associated overhead.

REFERENCES


Supeff aler allM8 )Q:.i to add the dimension of human speech output in your computer programs. Add voice to games. Program verbal prompting for the operator of your business system. Use verbal warnings under program control as an enunciator in commercial security or control rooms. Create educational programs that verbally coach the student.

THE SUPERTALKER SYSTEM.
SuperTalker is a new Mountain Hardware peripheral system which allows the Apple II computer to output exceptionally high quality human speech through a loudspeaker under program control. Output may also be directed through any P.A. or stereo system. Initially, spoken words are digitized into RAM memory through the system microphone. Speech data in RAM may then be manipulated like any other stored data.

A COMPLETE PACKAGE.
The SuperTalker peripheral system consists of:
The SuperTalker peripheral card which plugs into

FOR YOUR APPLE II

a loudspeaker; easy-to-use operating software and documentation; plus, two ready-to-run SuperTalker programs.

OPERATING SYSTEMS.
In order to achieve maximum utility using SuperTalker, the SuperTalker Disk Operating System permits output of human speech under program control with direct I/O routines. It also provides a preparation program which permits the creation of voice files on diskette. BASIC program routines are provided which require only one-line statements to output a word or phrase. Routines also support cassette storage.

TEACH YOUR COMPUTER TO TALK.
For $279 assembled and tested, SuperTalker gives your Apple II a voice in the matter.

AVAILABLE NOW.
Mountain Hardware's SuperTalker, Apple Clock and 100,000 Day Clock™ (for S-100 bus computers) are available through computer dealers worldwide.
MICROCHESS
The Industry's First
Gold Cassette
Over 50,000 Sold

MICROCHESS is the industry's best selling computer game. And originating from Personal Software. What's more, every Personal Software product is selected to give you these same benefits of easy availability, reliable cassettes, readable documentation, a user friendly interface, and most important, a guaranteed loading time after time. If you haven't already, order your own gold cassette: MICROCHESS, by Peter Jennings, for 8K PETs, 16K APPLES, and 4K Level I and II TRS-80s

TIME TREK
A Tour De Force
In Real Time Action
Strategy Games

TIME TREK by Brad Templeton for 8K PETs and Joshua Lavinsky for 4K Level I and II TRS-80s adds a dramatic new dimension to the classic Star Trek type strategy game: REAL TIME ACTION! You'll need fast reflexes as well as sharp wits to win in this constantly changing game. Be prepared—the Klingons will fire at you as you move, and will move themselves at the same time, even from quadrant to quadrant—but with practice you can change course and speed, aim and fire in one smooth motion, as fast as you can press the keys. Steer under power around obstacles—evade enemy shots as they come towards you—lower your shields just long enough to fire your phasers, betting that you can get them back up in time! With nine levels of difficulty, this challenging game is easy to learn, yet takes most users months of play to master. ADD SOUND EFFECTS with a simple two-wire hookup to any audio amplifier; the TRS-80 also produces sound effects directly through the keyboard case, to accompany spectacular graphics explosions! You won't want to miss this memorable version of a favorite computer game.

Where to Get It:
Look for the Personal Software™ display rack at your local computer store. If you can't find the product you want, you can order direct with your VISA/Master Charge card by dialing 1-800-325-6400 toll free (24 hours, 7 days; in Missouri, dial 1-800-342-6600). If you have questions, please call 617-783-0694. Or you can mail your order to one of the addresses below, as of the dates shown.

Until July 1: P.O. Box 136
Cambridge, Mass. 02138

After July 1: 592 Weddell Dr.
Sunnyvale, Calif. 94086

Personal Software™

BYTE July 1979

Circle 302 on inquiry card.
Sonic Anemometry for the Hobbyist

Neil Dvorak
2562 S Newport
Denver CO 80224

Meteorological measurement generally concerns itself with five variables: air pressure, humidity, temperature, wind speed, and wind direction. A single sonic anemometer can sense each of the last three variables. Accuracy and linearity are excellent. Additionally, the actual air temperature is detected in a manner which is insensitive to solar radiation, which can easily heat up conventional thermometers.

In a sonic anemometer, wind vane and rotating cups are replaced with transducers which measure the speed of sound as a function of wind velocity and temperature. Commercially available research grade instruments cost upwards of $10,000, and until recently employed analog computational circuitry.

The arrival of the microcomputer and its associated display techniques makes such a scientific instrument economically feasible as an experimenter’s project. As a bonus, data can be logged into memory over time, averaged, and displayed as desired. A tantalizing option involves the attachment of a fast but inexpensive 4 bit analog to digital converter which enables the instrument to double as an ultrasonic echo radar device. At this time, however, such investigations have progressed only to echoing observations on the time base of a triggered oscilloscope.

In operation the instrument uses a pair of pulse travel times in the North-South direction and a corresponding pair for the East-West direction. These vector components are easily processed into a resultant wind vector with magnitude and direction. Physically, two sets of ultrasonic transducers face each other at opposite ends of a path. Simultaneous sound fronts and eventual reception yield two travel times whose difference is a measure of wind speed along the path.

**Fundamental Relationships**

The following derivation yields wind speed:

\[ \Delta t = t_2 - t_1 = \frac{D}{C-W} - \frac{D}{C+W} \]

\[ = \frac{2DW}{C^2-W^2} \approx \frac{2DW}{C^2} \quad (1) \]

Therefore \( W = \left( \frac{C^2}{2D} \right) (\Delta t) \quad (2) \)

where \( C = \) speed of sound
\( D = \) path length
\( W = \) wind speed
\( t = \) difference of travel times.

The resultant wind speed, \( W_r \), being the sum of two orthogonal vectors, is simply expressed as:

\[ W_r = \sqrt{W_{NS}^2 + W_{EW}^2} \quad (3) \]

Temperature is found by adding a pair of travel times:

\[ t_1 + t_2 = \frac{D}{C+W} + \frac{D}{C-W} = \frac{2DC}{C^2-W^2} \]

\[ \approx \frac{2DC}{C^2} = \frac{2D}{C} \quad (4) \]

If \( C = 20 \sqrt{T_k} \) is substituted in the above relationship:

\[ T_k = \left[ \frac{2D}{20(t_1 + t_2)} \right]^2 \quad (5) \]

where \( T_k \) is degrees Kelvin.

Since the velocity calibration of the instrument varies about 3.5% over a 0°C to 30°C range, the temperature measurement can be used to correct the velocity readings. Using equations (2) and (4) above:

\[ W = \frac{2D(t_1 - t_2)}{(t_1 + t_2)^2} \quad (6) \]

Wind speed measuring resolution can be determined if the computer’s input cycle time and anemometer path length are known. Recall that:

\[ W = \Delta t \left( \frac{C^2}{2D} \right) \]
INTRODUCING — LOUMAR MANAGEMENT SYSTEMS SOFTWARE

4 programs to give you a comprehensive package for business management

The Loumar General Accounting System is a versatile, fully integrated software package designed for small and medium sized businesses. It is also suitable for CPA’s and bookkeeping service firms.

The complete software system is composed of four main modules: GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE AND PAYROLL. Each module may be used separately or in combination with any other module. Supplied on disk as run-time modules. Source not available.

All software is written in CBASIC II and utilizes the powerful CP/M operating system.

General system features include:
- Automatic posting to general journal
- Strict error detection
- Report production on demand
- Consistent operating procedures
- User oriented. No previous computer knowledge required
- Designed by accounting professionals
- Comprehensive, well presented reports and manuals
- Single or multiple client capabilities

HARDWARE SPECIFICATIONS

<table>
<thead>
<tr>
<th>Module</th>
<th>Specifications</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAYROLL</td>
<td>Up to 500 employees</td>
<td>$550</td>
</tr>
<tr>
<td>ACCOUNTS RECEIVABLE</td>
<td>Up to 1000 customers and 1000 monthly transactions</td>
<td>$550</td>
</tr>
<tr>
<td>ACCOUNTS PAYABLE</td>
<td>Up to 1000 vendors and 1300 monthly transactions</td>
<td>$400</td>
</tr>
<tr>
<td>GENERAL LEDGER</td>
<td>Up to 200 accounts with 2000 entries</td>
<td>$450</td>
</tr>
</tbody>
</table>

The end user’s microcomputer must satisfy the following requirements:

- 48k RAM
- Dual floppy disk system
- Printer with tractor. All printing is done in 80 col. format
- CRT with at least a 64 character by 16 line display
- CP/M and CBASIC II

Write for our brochure — Dealerships still available

Contact: Distributor

MISSION CONTROL • 2008 WILSHIRE BOULEVARD, SANTA MONICA, CA 90403 • (213) 829-5137

Circle 237 on inquiry card.
Assume a path length of 4.84 feet and \( C = 1100 \) feet per second. A Z-80 based computer operating at a system clock frequency of 2.5 MHz, for example, can accept input no faster than 8 µs per byte. Under these conditions, resolution is:

\[
8 \times 10^{-6} \text{ seconds} \times (1100 \text{ feet/second})^2 \quad \frac{2 \times 4.84}{2} \text{ feet} = 1 \text{ foot/second.}
\]

For physical construction convenience I chose a three foot path length. This gives a resolution of nearly 1 mile per hour.

**Construction Details**

Rigid support should be used for the transducer mounting. I chose 3/4 inch plastic pipe for low cost and ease of assembly. See figure 3 for more detail. The angle \( \alpha \) is not critical. Just keep in mind that the height (H) must be high enough to prevent off-axis energy from the transducers from bouncing off any hardware in the bottom center of the assembly (such as a printed circuit card). These reflections can return to the point of origin before arrival of the desired pulse. The plastic pipe for both axes can easily be fastened to a wooden base frame. If you need a permanent installation, you can investigate better support arrangements.

The ultrasonic transducers (Model MK-109) range from one to two dollars apiece on the surplus market, and they can be conveniently supported by pieces of 3/4 inch thin wall plastic pipe. Use shielded wire between the transducers and the interface electronics. The electronic circuit card should be sheltered from the elements. (A plastic sandwich bag will work for the short term.) Unshielded wire, such as a ribbon cable, can be used between the computer and the interface.

Mechanical adjustment, besides the obvious line-of-sight alignment, consists of physically moving one or more transducers in their holders so that both component vectors are zero in still air. A program such as the demonstrator routine in listing 1 should be used for this adjustment.

In wiring the preamplifier section of the receiver, note that all the 74C04 integrated circuits are connected to a separate 5 V zener regulated supply. The shields of the receiver-transducer coaxial cables connect to the negative side of this zener diode. This preamplifier common connects to ordinary digital ground at only one location — a lead from the negative side of the zener to supply ground.

**Interface Electronics**

Figures 1 and 2 may seem to indicate that considerable effort was wasted on obtaining an enormous signal to noise ratio. Not so. The barium titanate transducers (commonly used in intrusion alarms), having inherently high Q (ratio of inductance to resistance) and self resonance, are efficient only after many oscillations have built up. As impulse generators they are only marginally acceptable; I used them for their low cost and availability.

Complementary metal oxide semiconductor inverters, biased in their linear region, perform as stable high gain preamplifiers. The logic state edge detectors, formed by the comparator and type D flip flop combination, respond to the first negative or positive cycle received that exceeds a preset noise threshold.

The triacs are used to switch the output of the step-up transformer to either pair of transmitting transducers. Exclusive OR gate IC7c generates a delayed start strobe to the pulse generating circuitry. This delay allows the steering triacs to settle and permits only the desired set of transducers to activate. The monostable multivibrator IC10a, sensitive to transitions of either polarity, allows a single line from the output port to strobe the pulse generator and also select the desired wind direction to be measured.

The trigger threshold of all receiver circuitry is determined by a single resistor, \( R_t \), in a simple voltage divider string. \( R_t \) sets the difference between the comparator trip point levels, \( V_H \) and \( V_L \). The receiver must be sensitive enough to trigger on the second, third, or fourth incoming half cycle, but it must not be so sensitive as to latch up on extraneous noise. Increasing the value shown in the schematic, for example, decreases output sensitivity. Such action may be necessary if different path lengths or transducers are used.

Transformer T should have a turns ratio of approximately 10:1. A small 120 V to 12 V filament transformer will work here as a step up device, even at 40 kHz.

A precautionary note: when testing the pulse generating circuitry, do not run it continuously with transducers connected because the 200 V peak to peak signal could result in a burn-out of these devices.

**Software**

Listing 1 contains a program written for the Z-80 microprocessor which displays data from the anemometer. It sends data to one bit of an output port and accepts data from

Text continued on page 132
Figure 1: Schematic diagram of receiver section of circuit. The arrival time of a pulse is determined by the receiver as the first transition through the receiver threshold.
Figure 2: Schematic diagram of pulse-generation section of circuit. The pulse transmitter produces a very short 39 kHz tone burst to either of the two sets of transducers. The capacitors indicated by asterisks should be of temperature stable types such as paper or mica.
Table 1: Power table for the integrated circuits used in figures 1 and 2.

<table>
<thead>
<tr>
<th>Number</th>
<th>Type</th>
<th>+5 V</th>
<th>GND</th>
<th>-12 V</th>
<th>+12 V</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC1</td>
<td>74C04</td>
<td>14</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC2</td>
<td>74C04</td>
<td>14</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC3</td>
<td>74C04</td>
<td>14</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC4</td>
<td>MC3302P</td>
<td></td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC5</td>
<td>7474</td>
<td>14</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC6</td>
<td>7474</td>
<td>14</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC7</td>
<td>7486</td>
<td>14</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC8</td>
<td>MC3302P</td>
<td></td>
<td></td>
<td>12</td>
<td>3</td>
</tr>
<tr>
<td>IC9</td>
<td>555</td>
<td>8</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC10</td>
<td>74123</td>
<td>16</td>
<td>8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3: Structural diagram of the sonic anemometer transducer apparatus. Shown are the two transducers for one axis; only a cross section of the other axis is seen. The electronic circuit card is seen near the center of the assembly. The measuring resolution is a function of the distance D between the transducers. The angle α should not be less than 30 degrees.

Listing 1: Program in Z-80 assembler code to gather data from the sonic anemometer and display wind direction on a video monitor.

S 0600 0638: NOP NOP NOP CALL 0740:
0603 CD 40 07 CALL 0740:
0604 CD 00 22 CALL 0740:
0605 CD 55 07 CALL 0740:
0606 CD 50 07 CALL 0740:
0607 CD 50 07 CALL 0740:
0608 CD 50 07 CALL 0740:
0609 CD 50 07 CALL 0740:
0610 CD 50 07 CALL 0740:
0611 CD 50 07 CALL 0740:
0612 CD 50 07 CALL 0740:
0613 CD 50 07 CALL 0740:
0614 CD 50 07 CALL 0740:
0615 CD 50 07 CALL 0740:
0616 CD 50 07 CALL 0740:
0617 CD 50 07 CALL 0740:
0618 CD 50 07 CALL 0740:
0619 CD 50 07 CALL 0740:
0620 CD 50 07 CALL 0740:
0621 CD 50 07 CALL 0740:
0622 CD 50 07 CALL 0740:
0623 CD 50 07 CALL 0740:
0624 CD 50 07 CALL 0740:
0625 CD 50 07 CALL 0740:
0626 CD 50 07 CALL 0740:
0627 CD 50 07 CALL 0740:
0628 CD 50 07 CALL 0740:
0629 CD 50 07 CALL 0740:
0630 CD 50 07 CALL 0740:
0631 CD 50 07 CALL 0740:
0632 CD 50 07 CALL 0740:
0633 CD 50 07 CALL 0740:
0634 CD 50 07 CALL 0740:
0635 CD 50 07 CALL 0740:
0636 CD 50 07 CALL 0740:
0637 CD 50 07 CALL 0740:
0638 CD 50 07 CALL 0740:

Listing 1 continued on page 126
The latest Buyers-Guide of microcomputer software, accessories and supplies is now available. Up to the minute releases on software and accessories for the APPLE II and the TRS-80 as well as a wide range of computer supplies are listed on these sheets. This is the most comprehensive guide of its kind available today. If you own a microcomputer you can't afford to pass up this offer.

If you are a supplier of software or accessories you'll want a copy of this guide to see if your products are listed because if they aren't you are missing out on sales.

$3.00 (we will send you a certificate worth $3.00 off of your first purchase.)

AUTOMATIC INSTANT UPDATE

We send you the current Buyers-Guide bound in a handsome 3 ring binder. Monthly you will receive current updated pages to add to the catalog. A directory of suppliers, their phone numbers and their addresses are always at your finger tips. We publish user comments on items purchased. As a subscriber to Automatic Instant Update you will be the first to learn of new software releases, you'll get some great buys on overstocked items or new release specials, and you'll never have to search again.

One year's service $25.00

Wallace Electronics, Inc.
4921 N. Sheridan Rd
Peoria, Illinois 61614

SOFTWARE BUYERS GUIDE

Circle 385 on inquiry card.
The line your reputation should be on.

When a computer supplier or retailer sells equipment, his reputation is on the line.
And when the customer has to justify his investment decision, his reputation is on the line too.
So, naturally, everybody's a lot better off on a reliable line. Like ours.

With a name like Industrial Micro Systems, we wouldn't think of building anything short of industrial standards... even our S-100 products.

No short cuts or jumpers.
All of our boards—CPU, memory and controllers—are made from real fiberglass instead of a plastic substitute. And contacts are gold plated over a copper-nickel barrier.
Every disk enclosure features a heavy-gauge steel chassis, and a heavy duty power supply. Built to take it.
Then come our complete systems. Aside from the disk drives, we manufacture every component that goes into them. Even the desk that goes around them. So we can vouch for rugged reliability inside and out.

Two complete systems:
Our new Series 5000 features a megabyte of storage on integral mini-floppies. And our Series 8000 is all business too. It can handle up to three megabytes on 8-inch drives. A large library of software, growing daily, includes CP/M, PASCAL, and FAMOS*

Breaking the 64k barrier.
Using Industrial Micro Systems 32k memory boards with Memory Management, our systems can handle up to 576k RAM internally.

More in store.
That's not the end of the line. It's growing all the time, along with our reputation for quality. Put your reputation on the line. Our line.
For more details just call or write. Supplier/dealer inquiries welcomed.
Industrial Micro Systems, 628 N. Eckhoff, Orange, CA 92668. (714) 633-0355.

INDUSTRIAL MICRO SYSTEMS
The great unknown.

*Trademark of Digital Research **Trademark of MV/T Systems
Listing 1 continued from 126:

<table>
<thead>
<tr>
<th>Amount</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>08D9</td>
<td>CALL 08EB</td>
</tr>
<tr>
<td>08DC</td>
<td>08DF</td>
</tr>
<tr>
<td>08E1</td>
<td>CALL 08EB</td>
</tr>
<tr>
<td>08F7</td>
<td>08EA</td>
</tr>
<tr>
<td>08E8</td>
<td>08EB</td>
</tr>
<tr>
<td>08F1</td>
<td>08EA</td>
</tr>
<tr>
<td>08F2</td>
<td>08EA</td>
</tr>
<tr>
<td>08F3</td>
<td>08EA</td>
</tr>
<tr>
<td>08F4</td>
<td>08EB</td>
</tr>
<tr>
<td>08F5</td>
<td>08EB</td>
</tr>
<tr>
<td>08F6</td>
<td>08EA</td>
</tr>
<tr>
<td>08F7</td>
<td>08EA</td>
</tr>
<tr>
<td>08F8</td>
<td>08EA</td>
</tr>
<tr>
<td>08F9</td>
<td>08EA</td>
</tr>
<tr>
<td>08F10</td>
<td>08EA</td>
</tr>
</tbody>
</table>

08EB thru 08F4 counts the number of memory spaces necessary to find a bit set in the data.

0700 thru 0709 inputs a block of data from a port to a page in memory.

0710 thru 0728 places the N-S-E-W graticule in the buffer allocated for the video display.

0730 thru 073A is for delay only.

0740 thru 074F erases half the video buffer.

0760 thru 077D are the remaining video erase routines.

08A0 thru 08AB moves the asterisk to the left in proportion to the magnitude of the horizontal (E/W) wind vector. Limits the data to 15 decimal places to prevent overwriting other memory.
Get Ready to Make Your Own Magic

Announcement I. The first eight Personal Programs® from Aladdin Automation are waiting for you now at your neighborhood computer retailer or direct from Aladdin. Now you can get your full share of Aladdin magic in every one of these Personal Programs®.

Math-Ter-Mind® A delightful, educational learning experience for your pre-school child. Watch the smile on your child’s face as a correct answer makes the mathematician smile on the screen before you. A nursery song also serves as a reward for learning elementary addition and subtraction. With Aladdin’s Math-Ter-Mind® your child’s pathway to learning will be fun-filled...for both of you. Math-Ter-Mind® The first release from the Aladdin Education® Series. (nursery song currently available only on Apple II™ program)

Lunar Lander In a controlled descent, you’re just seconds away from your first landing on the cold, forbidding surface of the moon. As you navigate your delicate spacecraft downward to the safety of Moonbase, you must be ever watchful of the dangers rising to meet you with each passing moment: a fuel level fast approaching zero, deadly meteor showers that come from any direction, at any time, sheer-faced rock cliffs and rough terrain. Choosing the correct landing pattern and rate of descent Aladdin’s Lunar Lander. Your chance to reach out and touch the stars... without leaving the safety and comfort of your own chair. The first release from the Aladdin Simulation® Series

Craps All eyes in the casino are on you. The dice are in your hand. Lady Luck sits at your shoulder, whispering... Just one more time... Try your luck just one more time... You throw and watch the dice tumbling on the screen. With Aladdin’s Craps you play against the computer. So it’s awfully tough to win. But when you do... it’s an experience you’re likely never to forget. Craps. An exciting, heart-pounding Personal Program®. The first release from the Aladdin Las Vegas™ Series

Mastermind A challenging game of intrigue, centuries old, that will give you full chance to test your powers of logic, deduction and reason. And test them you will..as you try and solve the computer’s puzzle. using clues as they’re provided one-by-one. You control the degree of difficulty in this classic Personal Program® that offers one simple, yet all-consuming challenge beat the Mastermind in a direct, one-on-one battle of wits. Aladdin’s Mastermind. The first release from the Aladdin Old Favorites™ Series

Tic-Tac-Toe Five different levels of difficulty allow a person of any age or skill to take part in this relaxing, enjoyable game that can act as a learning tool, as well. Level I. for example. is suitable for children and is excellent also for teaching simple mathematics. The computer just about perfectly at Level V. Just about, that is, go ahead and take your best shot. See if you can beat the computer in this traditional favorite of young and old alike. Tic-Tac-Toe. Another first release from the Aladdin Old Favorites™ Series

Jungle Island™ Shipwrecked in a raging storm at sea, miraculously you survive only to find yourself stranded on a seemingly deserted jungle island. Without food, water or supplies of any kind, you begin to try and find your way to safety. The computer will be your eyes and ears as you explore your jungle island and all the mysteries and dangers that lie in wait for you. Jungle Island™ A captivating first release from the Aladdin Adventure™ Series

Stix* Aladdin’s Stix™ can be played with 2 to 5 piles of sticks and between 1 and 19 sticks in each pile. The object: to be the one to pick up the last stick. Sounds simple? Yes, but you’re playing against the computer. Take heart...for you can control the degree of difficulty in this update of the ancient game of Nim. Stix™ Another first release from the Aladdin Old Favorites™ Series

Super Pro Football™ Here’s your chance to be more than just an armchair quarterback. With Aladdin’s Super Pro Football™ you can replay any Super Bowl game, from the first, between Green Bay and Oakland, to last year’s classic victory by Pittsburgh over Dallas. For once you can turn back the clock and go for that one big play that made the difference between victory and defeat in pro football’s biggest game of all. Super Pro Football™ The first exciting release from the Aladdin Super Pro™ Series

Visit your neighborhood computer retailer or contact Aladdin direct to get your full share of the magic in Announcement I. the first eight Personal Programs® from Aladdin Automation.
Table 2: Summary of actions taken by program in listing 1.

1. Output to a port to trigger the North-South transducers.
2. Erase part of the video buffer (must be a time-invariant task).
3. Input a block of data into memory from anemometer port.
4. Examine memory, extract, and save North-South travel times.
5. Output to port to trigger East-West transducers.
7. Input a block of data into memory.
8. Examine memory, extract, and save East-West travel times.
9. Move cursor (equivalent to head of resultant wind vector) in video memory buffer appropriate horizontal (East-West) and vertical (North-South) distances from origin.
10. Display the contents of video buffer on the video monitor.
11. Compute and display the temperature. (optional)
12. Go to step 1.

Figure 4: Block diagram of anemometer hardware.

Figure 5: Timing relationships of signals present in the sonic anemometer.
Both our UNCommon Dynamic and our UNCommon Static RAMS have the following features and specifications:

- They are all GUARANTEED to be compatible with the following S-100 systems: CROMEMCO, IMSAI, ITHACA AUDIO, MITS, NORTH STAR, PROCESSOR TECHNOLOGY, TDL, TEL, VECTOR GRAPHICS, and other S-100 systems.

Both of Our UNCommon Dynamic RAM Series, the DMB-6400 and the DM-6400, feature:

- 64K bytes of dynamic RAM with up to 4MHz refresh rate.
- Memory interface compatible with crystal controlled timing INDEPENDENT of bus or processor timing.
- No wait states with 8080 or Z-80 to 4MHz. Up to 5MHz with 1 wait state.
- Memory selectable or deselected in 4K byte increments.
- Low power, 8 watts maximum, in 64K byte configuration.

Our UNCommon Static RAM Series, the SMB-3200, features:

- Memory Bank selectable capabilities: Either two (2) 16K byte banks of memory, or one (1) 32K byte bank per board.
- 32K bytes of low power static RAM.
- No wait states with 8080, 8085, or Z-80 processors up to 5MHz.
- Addressable in 4K byte increments at 4K boundaries. Deselectable in 4K byte increments.

Announcing the Model DMB-6400 Series of UNCommon Dynamic RAMS. This Series incorporates the features which are standard in the DM-6400 Series and adds the following capabilities:

- MEMORY BANK SELECT which is compatible with CROMEMCO NORTH STAR, and other systems using output port bank select.
- 256 ports selectable (including CROMEMCO 40H and NORTH STAR COH).
- Eight (8) 64K bytes banks of memory per output port.
- 16K bytes of addressable memory space may be individually set to start at 0000, 4000, 8000, or C000 and can be set for any of the 8 banks on one selected output port. The bank memory size can be incrementally from 16K bytes to 64K bytes in 16K increments, allowing 512K byte bank sizes.
- Total memory addressing to over 100 Megabytes by using different 1 O ports for memory control.

All of our UNCommon Dynamic and Static Series RAMS feature the following:

- Inputs are RC filtered and buffered with 1 LS TTL load. Compatible with terminated or unterminated buses.
- Outputs are all transistor open-collector.
- Disc compatible DMA compatible.
- Phantom memory selectable on pin 67.
- RIP switch selectable addressing.
- Reliability — all boards fully tested and burned-in.
- FULL DOCUMENTATION — Schematics, layout, parts list, theory of operation, timing diagrams, and option selection.
- Industrial quality design, components and construction. Glass epoxy boards, fused, double solder mask. Silk screened legends. Gold plated connector contacts. All IC's socket mounted.
- Guaranteed performance on parts and labor for one year.

For the above products see your local dealer or order directly.

ORDERS — BA. VISA, MASTER CHARGE, money orders, or personal checks. Please allow 14 days for checks to clear bank. California residents please add 6% sales tax to your order. All orders shipped postpaid. All orders in U.S. funds. Please add 10% on all orders outside the U.S.A. and Canada.

Circle 215 on inquiry card.
The interface electronic circuits are housed inside a convenient but necessary weather barrier.

Photo 2: This view shows the plastic pipe construction of the sonic anemometer and the positioning of the transducers.

Photo 3: The interface electronic circuits are housed inside a convenient but necessary weather barrier.

Text continued from page 122:

four bits of an input port. The composite wind vector is displayed on a video monitor in a 16 line by 64 character format. The program does not include a routine for computing the square root of the sum of squares; the plotting method used in the video formatting makes this unnecessary.

1 K bytes of memory is allocated for video display storage. A fixed cardinal direction graticule and an asterisk varying with the wind vector are placed in this buffer by subroutines. For each complete measurement by the transducers, the program erases and restores the video buffer. The position of the asterisk with respect to the center of the screen indicates wind direction and magnitude. Photo 1 shows the display. This arrangement, although cramping the display to only a ±8 unit variation in the North-South direction and allowing a ±32 unit variation in the East-West direction, still permits an interesting, virtually instantaneous display of wind activity.

Some of the subroutines in the listing have been split up so that the computer is doing useful things even during the transit times of the sound pulses (eg: erasure of the video buffer). The program measures the pulse travel times in a manner analogous to a counter. At a fixed time after pulse initiation, the input routine begins to look at a particular pulse has arrived flip flop (part of the hardware interface that connects to an input port) and records its logic state, 0 or 1, into an initial location in memory. This routine repeats itself 256 times, each time entering another observation into the next memory location allocated for logging. Checking for a 1 state (pulse has arrived) is saved for later in order to get the best time resolution out of the processor. A number whose value is proportional to travel time is finally obtained by a routine which starts at the beginning of the logging buffer and counts the number of successive memory bytes necessary to find a 1 in a particular bit position. Since a bit is allocated for each of the four travel times, determining the wind direction and speed is simply a matter of testing each of the four bits in an identical fashion.
This is a thoroughly tested and successfully installed software package. It is very user oriented and simple to use. The package is as comprehensive as available computer tax services. It will calculate taxes, prepare and print all forms.

This package is supported by American Tax Associates, an established California accounting firm. In this way you can be assured that the yearly updates will be consistent with the current laws and accounting practices.

This package is a real time saver. It can perform income averaging automatically, and based on the data input, the program can determine whether to itemize or to use the standard deduction.

The client data collection and input procedures were selected based on the experiences of American Tax Associates, and the techniques used by many service companies. A simple form is completed during the client interview. The data from this form is later input into the computer for processing.

When the client data is entered into the computer you may select to have it print an audit trail of all data entered. This will enable you to double check the data entered.

The returns are printed on continuous preprinted IRS approved forms. Those forms not requiring a preprinted form are formulated and printed on blank paper. The data disk will hold up to 120 clients so the software is designed to print all of one page at a time.

The Alpha Micro system was chosen as the base computer system because of its multiuser capability, high throughput, and upward expandability into a hard disk system.

Yearly updates will be supported by American Tax Associates. These updates are available from either your dealer or directly from Mission Control.

SYSTEM REQUIREMENTS:
- Language: Alpha Micro Systems Basic (compiled)
- Media: 8' floppy diskette
- CPU: Alpha Micro AM-100
- Memory: 64K RAM
- Printer: 132 col with tractor feed
- Floppy: Dual 8' drives required
The Nature of Robots

Part 2: Simulated Control System

In part 1, we went through a chain of reasoning that ended with the conclusion that the behavior of an organism is not what it seems. Behavior appears to be at the end of a cause and effect chain that starts with the inputs to a nervous system, but that chain is subject to disturbances that can occur after the output of the nervous system. Nevertheless, the behavior at the end of this chain is stable and repeatable, while events closer to the organism become less predictable as we get nearer to the neural signals at the output of the nervous system. By analyzing an example in which a car is maintained in the center of its lane, we saw that this measure of behavior belongs at both the cause and effect ends of the chain, and that if this variable is shown only once in the diagram, a closed loop results.

We are going to look in more detail at the behaving system in this closed loop, to see how it might be organized to produce the results seen. We will start using a simulator written in BASIC which allows the user to vary many parameters of the control system to see the effects on its actions. Human behavior will not be mentioned much in this installment; there are many fundamentals to cover before we can get back to the main purpose of this series. The object here is to retrain the intuition so that the closed loop way of seeing behavior becomes as natural as the old straight through cause and effect way.

Organization of a Control System

The simulator (listing 2) is set up to demonstrate the properties of a standard sort of control system organization. We will first look at that organization, then at the simulator itself, and finally at some details of the operation of the control system. You
can do much more experimenting than we will discuss here.

Figure 5 is a diagram of a typical control system. Almost every control system can be expressed in this form, although in the real world, both the output quantity and the disturbing quantity may have many effects other than those on I, but those effects are irrelevant to the operation of this system (perhaps not to the designer or user of the system, if it is artificial). We have therefore considered everything about the environment that is of interest here.
Figure 5: The system's output quantity, \( Q \), influences the input quantity, \( I \), via the feedback function, \( FNF \). The disturbing quantity, \( D \), influences the input quantity via the disturbance function \( FND \). Both \( FNF \) and \( FND \) represent physical links in the environment. The state of the input quantity is determined by the sum of these two influences.

The system's input function, \( FNI \), converts the state of the input quantity into a magnitude of the perceptual signal \( P \). \( P \) is compared with the reference signal \( R \) in the comparator function, which emits an error signal \( E = R - P \). The error signal is converted into a magnitude of the output quantity via the output function, \( FNO \).

Above the line we have the behaving system. We cross the boundary at the input function, \( FNI \). This is the function which turns the state of an external quantity, \( I \), into the magnitude of a perceptual signal, \( P \). Both sensors and computing processes may be involved in a complex input function. The outcome, however, is always the magnitude of a single signal, whatever it represents. This signal can only increase or decrease; we will always work with one-dimensional control systems, treating multidimensional control phenomena by using multiple control systems. The perceptual signal is the system's internal representation of the external world — its only such representation.

Line 103 expresses the definition of the input function and the way it relates the input quantity and perceptual signal: \( P = FNI(I) \).

Inside the system is another signal, the reference signal, \( R \). In living systems, this signal is generated elsewhere in the organism; it is not accessible from outside. The reference signal, along with the perceptual signal, enters a function called the comparator, which subtracts one signal from the other and emits an error signal, \( E \), representing the signed difference of magnitudes. It does not matter which signal is subtracted from which, but for uniformity we will always treat the reference signal as the positive input and the perceptual signal as the one subtracted from it. Thus, a positive error signal always means that the reference signal is larger than the perceptual signal. This function does not have to be generalized, as nonlinearities and amplification can always be absorbed into one of the other functions.
FREE SOFTWARE!

PRACTICAL MICROCOMPUTER PROGRAMMING: THE Z80
by W.J. Weller

Here from W.J. Weller and Northern Technology Books is the third in the Practical Microcomputer Programming series. It is a comprehensive text covering assembly language programming for Z80 based microcomputers. The first 16 chapters cover Z80 programming comprehensively, from binary operations to interrupt handling. Included are chapters on moving data, logical and arithmetic operations, use of the stack, communications with the terminal, floating point arithmetic and graphic output. All programming techniques are illustrated with formal tested examples. An important feature of the book is that it uses the universal standard 8080 mnemonics. This is of great help to users who are upgrading their machines and software to utilize the Z80 processor.

The last part of the book is software; an editor/assembler which will run on any 8080 or Z80 machine and a debugging monitor.

Hardcover $29.95

PRACTICAL MICROCOMPUTER PROGRAMMING: THE M6800
by W.J. Weller

This second volume of the Practical Microcomputer Programming series addresses the problems of applications programming at assembly level for the M6800. In 16 chapters and more than 100 formal examples, the fundamental techniques of assembly level programming are applied to the solution of specific problems with the 6800. Nowhere theoretical, it is a thorough and detailed methods text for the beginning and intermediate application programmer using the 6800.

$21.95 hardcover.

* A trademark of Tandy Corporation

FINALLY....

FREE:

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 basis or can be made to do so by buffering. The user supplies his or her 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 usable 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 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". $14.95

- 8080/8085 editor/assembler object code on diskette for North Star disc systems. Only $14.00
- 8080/8085 editor/assembler and debug source code for North Star disc systems. Only $14.00
- Or both for $26.00!

PRACTICAL MICROCOMPUTER PROGRAMMING: THE INTEL 8080
by W.J. Weller, A.V. Shatzel and H.Y. Nice

Here is a comprehensive source of programming information for the present or prospective user of the 8080 microcomputer, including moving data, binary arithmetic operations, multiplication and division, use of the stack pointer, subroutines, arrays and tables, conversions, decimal arithmetic, various I/O options, real time clocks and interrupt driven processes, and debugging techniques.

This 306-page hardcover book is well worth its $21.95 price and should be in every 8080 or Z80 user's library.

PRACTICAL MICROCOMPUTER PROGRAMMING: THE M6800
by W.J. Weller

This second volume of the Practical Microcomputer Programming series addresses the problems of applications programming at assembly level for the M6800. In 16 chapters and more than 100 formal examples, the fundamental techniques of assembly level programming are applied to the solution of specific problems with the 6800. Nowhere theoretical, it is a thorough and detailed methods text for the beginning and intermediate application programmer using the 6800.

$21.95 hardcover.

FREE:

Send in the coupon supplied with the book and receive FREE the object programs of the editor/assembler and debug on either paper tape or on cassette tape for the TRS-80 microcomputer. (Loads in Level I with 16K RAM or Level II.)

Both the book and software for only $29.95!

- Also available is the editor/assembler object code for the Z·80 on diskette for North Star disc systems. Only $14.00
- In addition, the editor/assembler and debug source code for the Z·80 are available on diskette for North Star disc systems. Only $14.00
- Or both for $26.00!

BITS inc
Books to erase the impossible

Please send me the items checked below.

☐ Practical Microcomputer Programming — The Z80
☐ Practical Microcomputer Programming — The M6800
☐ Editor/Assembler System for 8080/8085 Based Computers
☐ 8080/8085 editor/assembler object code for 8080/8085 on diskette for North Star disc systems
☐ 8080/8085 editor/assembler object code for the Z·80 on diskette for North Star disc systems
☐ Editor/assembler object code for the Z·80 on diskette for North Star disc systems
☐ Editor/assembler object code for the Z·80 on diskette for North Star disc systems
☐ SAVE $2.00! Both object code and source code for the Z·80 (with choice of either FREE paper tape or FREE TRS-80 cassette when I mail coupon supplied with book) $29.95
☐ SAVE $2.00! Both object code and source code for the Z·80 (with paper tape object copies of the software sent FREE when I mail coupon supplied with book) $29.95
☐ SAVE $2.00! Both object code and source code for the Z·80 (with choice of either FREE paper tape or FREE TRS-80 cassette when I mail coupon supplied with book) $29.95
☐ SAVE $2.00! Both object code and source code for the Z·80 (with paper tape object copies of the software sent FREE when I mail coupon supplied with book) $29.95

Please send me the items checked below.

☐ Check Charge Card ☐ Master Charge ☐ Visa

Card #_________________ Signature_________________

Name__________________________
Address________________________

City______ State____ Zip__________

Expires______

Dial your charge card orders toll-free: 800-258-5477 Mon.-Fri 9-5 PM (In New Hampshire dial 924-3355)

You may photocopy this page Dealer inquiries invited
Anatomy of the Simulator

Let's run through the simulator quickly before we start using it, to see how this control organization operates.

Lines 1 thru 16 are user instructions. Lines 17 thru 27 initialize the system in a way that will be used to illustrate a point. Lines 28 thru 33 do more initializing, and ask for the width of your display. Lines 34 thru 36 create a blank string in case your BASIC doesn't set dimensioned strings initially to spaces.

Lines 37 thru 46 define the various functions of the control system. If your BASIC can't do multil ine functions, you can substitute subroutines here. The idea is to make it easy to try out different kinds of functions in the control system.

Lines 49 thru 91 comprise the interpreter, which accepts character strings and sets initial conditions and parameters before each run. Variables are initialized and constants are set by typing a string of the form A=m or A=n-m (no spaces; terminated by a carriage return). To set up the plotter, the statement is PLOT XXXXXXX, where XXXXXXX is one or more characters from the set P,E,R,I,O, and D, in any sequence. The plotter comes up set to plot P, E, R, and I. If you forget the last values of the parameters K1, K2, S1, and S2, type 1 and they will be printed out. We will eventually define them.

The control system itself is simulated from line 95 to line 108. Entering the simulator at line 95 initializes the perceptual and output variables to values given to the interpreter. Entering at line 99 runs the simulation from the conditions left at the end of the last run. This is taken care of by the two run commands in the interpreter: a dot (.) means run with initialization, and a slash (/) means run without initialization. All commands require a carriage return termination.

The plotting subroutine goes from line 112 to line 128. Its operation deserves a note, since it was arrived at after some more normal schemes were rejected for being too slow. When the interpreter is given a string of symbols to set up the plotting, a table is set up (V(j)) in which a 1 means plot and a 0 means don't plot. When the plotter is entered, it transfers all six variables to another table, U(j). The output buffer is then cleared, and a short loop scans the V table, picking up variables from the U table when V(j)=1, and putting the symbol into the output buffer in a position corresponding to the value of the variable. Then the output buffer is printed out. This eliminates sorting the variables by size or printing the line as many times as there are variables. This method nicely cures the fundamental "rheumatism" of BASIC, as it is able to plot about two lines per second on my Polyomatics VTI display.

When two variables fall on the same spot, the variable that actually appears is the latest one in the series PERIOD. Thus far is has always been easy to figure out where a missing variable is hidden.

Once we have a set of variables connecting functions together, and an overall arrangement, we can treat the system by assembling it piece by piece. Let's look at the pieces we have, represented by the four statements in listing 2 from line 102 to 105:

\[
\begin{align*}
102 & \quad I = FNF(O) + FND(D) \\
103 & \quad P = FNI(I) \\
104 & \quad E = R - P \\
105 & \quad O = FNO(E)
\end{align*}
\]

Looking at figure 5, we can see that these four statements lead us clockwise around the closed loop. I is the result of combining the outputs of the feedback and disturbance functions. It becomes the input to the input function, producing a value of the perceptual signal P. P is one of the inputs to the comparator, which produces the error signal E.

Continued on page 140

Therefore line 104 represents the comparator without using a function; it is the comparator function itself: \( E = R - P \).

The error signal drives the output of the system via the output function, FNO. The output of the system, therefore, depends not on the input quantity or the perceptual signal alone, but on the difference between the perceptual signal and the reference signal. The output function translates a signal inside the system into a quantity outside it, according to whatever rule is described by FNO. If the error signal changes sign, the output quantity also changes; in other words, we assume that output functions have no constant term. Any such constant term would have the same effect as a reference signal, creating an offset in the overall system response. Not every system can handle error signals and output quantities that go through zero and thus change sign, but the principles remain the same in the region where the system works.

Line 105 expresses the operation of the output function: \( O = FNO(E) \). This closes the loop of cause and effect since the output quantity appears in line 102 where the input to the system is calculated.

If the system functions are properly designed for the properties of the system's environment, this entire closed loop will seek an equilibrium state. Our simulator will let us look at time-varying effects, but for the most part we will be concerned with steady state relationships.

Once we have seen how time variations come into the picture, we will concentrate on variations that occur slowly enough that the system and its environment never get far from a steady state relationship. This is the whole trick in grasping how control systems work. If you allow yourself to become embroiled in the interesting details of stabilization, or interested in the limits of performance in the presence of large and rapidly changing disturbances, you may learn a lot about one control system, but you will miss the organizational features that are obvious only when the system is not being subjected to unusual stresses. We will be concerned mainly with the normal range of operation, the range within which this system can behave very nearly like an ideal control system. Once that mode of operation is understood, there is plenty of time to explore the limits of operation. (See "Anatomy of the Simulator" text box).

A Wrong Approach

Let us start off by assuming that we have a simple linear system. The input function is a multiplier of 1, the comparator is already
WHY NOT...

JUST WRAP

- AWG 30 Wire
- .025" Square Posts
- Daisy Chain or Point To Point
- No Stripping or Slitting Required
- JUST WRAP™
- Built In Cut Off
- Easy Loading of Wire
- Available Wire Colors: Blue, White, Red & Yellow

*$14.95

MACHINE & TOOL CORPORATION 3455 CONNER ST., BRONX, N.Y. 10475 (212) 994-6600/TELEX 125091

*MINIMUM BILLING $25.00 / AD SHIPPING CHARGE $2.00 / NEW YORK CITY / STATE RESIDENTS ADD APPLICABLE TAX.
E is the input to the output function that produces Q, the output quantity. The output quantity is the input to the feedback function, which leads us back to the start.

It might seem that all we have to do now is to supply some specific forms for the functions, and turn the system on to see what it will do. In a sense, this is right. If this were an analogue computation, we might even get a correct idea of how the system works. However, it is unlikely that anyone who hasn't done this before would plug in the right functions to make a digital computer give us anything more than a fairy tale. It is so important to understand this point that I have written the simulator to come up initialized in order to illustrate it.

In fact, that you do solve it (by successive substitutions). Solve for the value of the perceptual signal in terms of R and D. You'll get P = R - 0.8 x D/2.

Ready for a shock? Your computer can't come up with that solution! Let's fire up the BASIC simulator, which is initialized according to equations 1 thru 4 above, and plot I, D, and O. Type RUN, and answer the question with a reply that tells the width of your display. After the colon prompt appears, type in the following:

```
RUN
```

simple and linear, the output function is a multiplier of 2, the feedback function is a multiplier of 0.5, and the disturbance function is a multiplier of 0.8. These choices are dictated partly by the need to keep variables from falling on each other when we plot them. The simulator initializes D to 15.

Our four system equations, with these values substituted, now look like this:

\[
\begin{align*}
I &= 0.5xO + 0.8xD = 0.5xO + 12 \quad (1) \\
P &= I \quad (2) \\
E &= R - P \quad (3) \\
O &= 2xE \quad (4)
\end{align*}
\]

This system of equations is iterated during a simulation of behavior.

The above is a pretty simple system of equations. So why can't we just solve it algebraically and skip the rest? I suggest, in fact, that you do solve it (by successive substitutions). Solve for the value of the perceptual signal in terms of R and D. You'll get P = R - 0.8 x D/2.

I trust nobody had trouble with that.

The dot says "do a plotting run after initializing the variables." A slash (/) would say "do the run from where the last run left off." The result can be found in figure 6.

The disturbance is set to a steady +15 units, and the reference signal is initialized to 0. According to the algebraic solution above, the input signal should be a steady 0.8 x 15/2, or 6 units, to the right of center (dots indicate center when nothing is there). It is clear that something else happened. The whole system is in a state of endless oscillation. (When variables fall on top of each other in a plot, the visible one is the latest in the sequence PERIOD.)

Nature has a way of slapping your wrist when you forget something important. Our wrist has just been slapped. Naturally we do not get the same result that algebra gives: the algebraic solution comes from treating all of those relationships simultaneously.

Our computer program is treating them one at a time. The algebra says that if one variable changes, they all change. The computer, being a purely sequential machine, thinks it can change one variable without changing the others. If the physical system being modeled is of that nature -- if it, too, is a sequential state machine -- then the computer will produce a correct picture of behavior. But, if the system being modeled works in terms of continuous variables, even in part, the computer will turn it into a sequential-state machine and analyze that kind of system instead of the one we actually have. That is what has happened here.

We forgot to tell the computer that these variables can't change as fast as the computer can compute.

A More Accurate Approach

In order to make this simulated system behave the way the algebra says it should, we have to slow down changes in one or more variables to take account of the fact that we are dealing with real, physical variables.

Figure 6: The initial plot generated by the BASIC simulator. Disturbance is set to 15 units and the reference signal is initialized to 0. The system is in a state of oscillation.
ONE PACKAGE DOES IT ALL
Includes these Application Programs . . .
Sales Activity, Inventory, Payables, Receivables, Check/Expense Register,
Library Functions, Mailing Labels, Appointments, Client/Patient Records

RANDOM, MULTI-KEY RECORD RETRIEVAL under CP/M, CDOS, IMDOS, ADOS . . .

SELECTOR III allows instant recall of any record using any information item in the record. That statement deserves re-reading, because that ability makes SELECTOR III the most powerful Date Base Management System in microcomputers today!

With SELECTOR III you can:
• define a record format, assign retrieval keys, and begin entering data in minutes.
• create sorted pointers to records matching your specific or range of requirements.
• automatically generate reports with control-break summaries and unlimited variety.
• bring an application online in hours instead of months.

SELECTOR III comes complete with eight application programs that perform the tasks listed at top of page. And, since it's distributed in source code form, you can easily add subroutines to do specific computations or file updates.

SELECTOR III runs under CBASIC Vers. 1 or 2, and is priced at $295. SELECTOR III-C2 is dedicated to Vers. 2 only, runs about twice as fast, and costs $345.

Both systems are available in a variety of CP/M, diskette size and density formats including IBM 8"; North Star; Micropolis; TRS-80; Processor Tech Helios II; Altair; iCOM; Dynabyte; Imsai; and others.

Available from computer stores nationwide:
LIFEBOAT Associates
2248 Broadway, Suite 34,
New York, N.Y. 10024 • (212) 580-0082

Or order direct from
MICRO-AP
9807 Davona Drive, San Ramon, CA 94583
(415) 828-6697
The slowing factors have been changed. S1 equals 0.5 and S2 is 0.2. We now have a much smoother curve.

and not abstract numbers. The simulator does this in the input and output functions, lines 37 thru 40 (input) and 41 thru 44 (output). We will be basically dealing with a linear system in which both the input and output functions are constants of proportionality. As you can see from listing 2, however, there's a little more to it than that.

Consider line 42: \( O = O + S2 \times (K2 \times E - O) \). The \( O \) on the left side is the new value of that quantity after this program step has been executed. On the right side, \( O \) indicates the last value of the output quantity. We recognize \( K2 \times E \) as a calculation of the output quantity as if it were simply proportional to the error signal, \( E \). The expression in parentheses, therefore, is the difference between this calculated new value and the old value of \( O \). This is how much the output quantity would change if it could change instantly.

This calculated amount of change is multiplied by \( S2 \), a slowing factor, and the result is added to the old value of \( O \). We calculate the amount of change that an instantly reacting system would produce, but allow only a fraction \( S2 \) of it to occur on any one iteration. \( S2 \) is a positive number between zero and one. We've put a low-pass filter into the output function, without affecting the steady state proportionality constant.

The same thing is done for the input function. A slowing factor \( S1 \), between zero and one, acts to slow \( P \) down. We need only one slowing factor to make this simulator behave realistically, but there is provision for two, so that you can explore the effect of having two if you wish. In all the plots to follow, we'll use a modest slowing factor of \( S1=0.5 \) in the input function, and essentially all of the required slowing in the output function. Once you get the hang of this you can put slowing factors into any of the functions.

The simulator is initialized with \( S1 \) and \( S2 \) set to 1, which reduces \( O + S2 \times (K2 \times E - O) \) to \( O + K2 \times E \). \( O \) or just \( K2 \times E \) (no slowing at all). The same is done for the input function. Let's set them to other values and see what happens. The values of \( S1 \) and \( S2 \) can be set by typing \( S1=n \) or \( S2=n \) and a carriage return:

\[
: S1=0.5 \\
: S2=0.2 \\
:. \\
\]

Suddenly we see nice, smooth relationships (figure 7). If you measure, you'll see that the input signal, \( I \), ends up just six units to the right; the same solution given by the algebraic approach.

Does this mean we can just use algebra to analyze a control system? Not at all. We won't delve into this, but the algebraic solutions are valid only if the differential equations which really describe the system have steady state solutions. Then the algebraic solutions are the steady state solutions. In our simulator, we see all the time variations that lead toward the steady state, and the algebra says nothing about these. By putting the slowing factors into our calculations we have caused this system to seek a steady state. Therefore, it is the stability of the system that tells us we can use algebra, not the other way around. Predicting stability can become a messy process. We fiddle around with slowing factors until we get stability, which is more or less how Nature does it anyway.

We have now established the fact that using natural logic and following causes and effects around the closed loop as a sequence of events will lead to a wrong prediction of control system behavior. This immediately eliminates three-quarters of what biologists, psychologists, neurologists, and even cyberneticians have published about control theory and behavior. We are just beginning to see that one must view all the variables in a control system as changing together, not one at a time. This is what I mean by retraining the in-
BATTERY SUPPORTED
CALENDAR CLOCKS

PDP-11*
TCU-100 • $495
- Provides month, day, hour, minute and second.
- Can interrupt on date/time, or periodic intervals.
TCU-150 • $430
- Provides year, month, day, hour, minute and second.
- Automatic leap year.
- Patches for RSX-11M, RT-11 FB/SJ VO2, VO3 and UNIX.

LSI-11/2*
TCU-50D • $295
- Provides month, day, hour, minute and second.
- Dual size board.
- Patches for RT-11 SJ/FB VO2, VO3B.

Lockheed SUE
TCU-200 • $550
- Provides year, month, day, hour, minute, second and milli-second.
- Interval interrupts between 1/1024 seconds and 64 seconds.

Computer Automation (Naked Mini)
TCU-310 • $385
- Provides year, month, day, hour, minute and second.

Multi-Bus**
TCU-410 • $325
- Provides year, month, day, hour, minute and second.
- SBC/BLC compatible.

HP 2100
TCU-2100 • $395
- Correct time restored after power failure.
- Compatible with the HP TBG card.

Serial Clock (RS 232 or 20 mA)
SLC-1 • $575
- Connects between any terminal and host computer.
- Provides date, time and more!

All Digital Pathways TCUs have on board NICAD batteries to maintain time and date during power down. Timing is provided by a crystal controlled oscillator. Prices are U.S. domestic single piece. Quantity discounts available.

For more information on these products, contact:
Digital Pathways Inc.
4151 Middlefield Road
Palo Alto, CA 94306
Phone: (415) 493-5544

*Trademark of Digital Equipment Corporation
**Trademark of Intel Corporation
Using the Simulator

The simulator is run from the keyboard, using commands that tell it which variables to plot and what values of variables and parameters to start with. The instructions can be given one at a time, terminated by carriage returns, or they can be given in a continuous string with commands separated by commas. The latter is useful for altering parameters in the middle of a plot in order to see their effects.

The only time a space is permitted in a command or string of commands is when it is separating the word PLOT from the string of variable symbols to be plotted.

In order to tell the simulator what variables to plot, type:

```
PLOT XXXXXX
```

where XXXXXX means a string of 1 to 6 symbols from the set PERIOD. The order of the symbols makes no difference. When two or more symbols land on the same plot, the one that you see is the latest in the series PERIOD, regardless of the order in which they were given.

To start a plotting run, type a period followed by a carriage return or comma if initialization is to occur first, and type a slash (/) if the run is to start from the conditions at the end of the previous run. Initializing creates one extra line of plot showing the initial conditions.

The parameters and variables that can be set are as follows:

- **L** Number of lines to be plotted in any plotting run.
- **K1** Steady state proportionality factor of the input function.
- **S1** Slowing factor for the input function; positive and between 0 and 1.
- **K2** Steady state proportionality factor of the output function.
- **S2** Slowing factor for the output function; positive and between 0 and 1.
- **O** Initial value of output quantity.
- **P** Initial value of perceptual signal.
- **R** Setting of reference signal.
- **D** Magnitude of disturbing quantity.

Examples: (colon is prompt from computer. Always terminate a string with a carriage return).

```
Set L to 16 :L=16
Set D to 0, run without initializing :D=0, or
                      :D=0
                   :/
Set D to 0, plot 2 points after initializing, set D to
      PLOT PER,D=0,L=2,,D=10,L=13,/
     10, plot 13 points from
            previous conditions. Plot P,E, and R
```

The program is written so that after a plot is completely done (a complete string has been interpreted), the prompt character appears to the right without a carriage return. That allows a 16 point plot to be shown on a 16 line video display screen without the final carriage return bumping the first line off the screen. If you want your next string to start at the left, just hit a carriage return.

To find out the values of K1, K2, S1, and S2 when you forget them, type "?" followed by carriage return and they will be printed.

Properties of a Control System

Figure 8 shows the control system and its environment as we will be dealing with it from now on. Let's start with some definitions:

**Loop Gain** means the product of all the steady state factors encountered in one trip around the closed loop, counting the comparator as a factor of -1. In the initial setup, K1 was 1, K2 was 2, and the feedback function FNF was a multiplier of +0.5, so the loop gain was -1. The sign of the loop gain is the sign of the feedback; we have (and will continue to have) negative feedback.

**Error Sensitivity** is the factor K2, the steady state proportionality factor in the output function FNO. This number expresses how much output will be generated by a given amount of error signal.

**Input Sensitivity** is the factor K1, the steady state proportionality factor in the input function FNI. This number expresses how much perceptual signal will be generated by a given amount of input quantity.

We are going to perform a series of experiments with this control system in order to arrive at some useful rules of thumb for thinking about how control systems work. These rules are approximations, but by doing the experiments and seeing how good the approximations are, you will learn to think precisely about control phenomena, even when using approximate language.

We will set the system parameters to give a loop gain of -10. As a way of summarizing where we are (refer to figure 8), the commands are given one at a time with annotations:

```
:K1=1 Input sensitivity = 1.
:K2=20 Error sensitivity = 20.
:S1=0.5 Input slowing factor = 0.5.
:S2=0.07 Output slowing factor = 0.07.
:R=0 Reference signal = 0.
:O=0 Output initialization = 0.
:P=0 Perception initialization = 0.
:D=0 Disturbance = 0.
```
DIABLO PROVES LOOKS ARE EVERYTHING.

With Diablo's printers and terminals, you can always be sure that beauty will be in the eyes of the beholder. Because no one knows more about print wheel technology than the company that invented it in the first place.

Diablo's metal and plastic wheel printers have established industry standards for crisp, clear characters, proportional spacing, and uniform density.

So, when you're ready to choose a printer for your own computer, pick the one that produces "picture perfect" originals every time.

If you really want to look good, remember this. With Diablo, you'll always look your best.

Diablo Systems

Diablo® is a registered trademark of XEROX CORPORATION.
Type those commands, and the system is now set up in a "home base" condition. Remembering that the comparator is equivalent to the factor of \(-1\) and the feedback function is permanently set to be a factor of \(+0.5\), this combination of parameters gives a loop gain of \(1 \times (-1) \times 20 \times 0.5 = -10\). There are two fundamental rules of thumb: a control system keeps its perceptual signal matching its reference signal, and the output of a control system cancels the effects of disturbances on the input quantity. We will take these up in order.

Rule 1: \(P = R\)

We're looking at the system with no disturbance acting \((D=0)\). If you want to be sure that everything stays at zero, type PLOT PERIOD, followed by a carriage return. You will see a row of Ds, D being the last symbol in the sequence PERIOD and hence the only one visible when all variables are at zero.

Now we will plot just the reference signal and the perceptual signal. The first two points will be done with the initial conditions set up above. The reference signal will then be set to +25 units, and the plot will be continued for 13 more points. Since this plot will commence with initialization (the dot command), an extra line showing the initial conditions will be plotted first. This makes a total of 16 lines, which will fit on most video displays. Of course, if you're doing this on paper you don't have to worry about the number of points plotted. Here is the command string:

---

**WE HAVE IT**

What is IT? The New Pascal Microengine.

IT features the new Western Digital 16 Bit Pascal Microengine CPU, 64K RAM, (2) RS232 ports, (2) parallel ports & (2) Shugart 8014 dual density drives. IT will directly execute Pascal generated P-Code (the only CPU on the market designed to directly execute a high level language), & IT will run compiled BASIC. Wait, that's not all — IT normally sells for $4,495, but you can have IT for $4,195. Dealer prices are much lower. Plus we have software galore.

Call us about IT at (803) 756-6000.

THE PASCAL MICROENGINE
FROM THE COMPANY THAT GIVES YOU TOMORROW'S INVENTIONS TODAY

---

![Figure 8: Adjustable parameters are K1 (input sensitivity), S1 (input slowing factor), K2 (output sensitivity), and S2 (output slowing factor). P and O can be initialized to any starting value (normally zero). R and D can be set, and remain the same during a run. The value of the feedback function is set at 0.5, the value of the disturbance function at 0.8.](image-url)
BIZ-80
The Business Software People

Mail List II by BIZ-80 Complete mail list system for dual disk. Enter, merge, sort, and print mailing labels. D, 32K $99.95

Small Business Bookkeeping by Roger W. Robitaille, is based on the Dome Bookkeeping Journal, sold for years in stationery and discount outlets. Level II, 4K with ($22.00) or without ($15.00) Dome journal.

Inventory System II by BIZ-80 Proper inventory management is the backbone of a profitable business, yet it's very difficult to keep current on price increases, shrinkage, low-on-stock items, profitable items versus losers, without an efficient and prompt method of surveying your inventory levels at any given time. This program can help you to achieve effective management — it can handle up to 1,000 items on one disk; each additional disk can handle another 1,000 items. With Documentation, $150.00

Inventory I I .2 Disk based program allows for creation, maintenance and review of over 2,000 items per clean diskette. Operates under Disk BASIC, $19.95.

ST 80 - SMART TERMINAL
Lance Micklus

 Turns your TRS-80 into a computer terminal. Features include: LOCK/CONTROL key, REPEAT key, ESC key, RUN key and a functioning BREAK key. Lets you list incoming data on line printer. Reprogram RS-232-C switches from keyboard, ESC key, RUN key and a functioning BREAK key. Level I or II, 16K $25.00

Inventory System II

APPENDIX

SoFTSIDE Your BASIC software magazine

Regular 1 year subscription - 12 issues - $15.00

PROG/80 For the serious programmer, from beginner to professional 1 year regular subscription - 4 issues - $10.00
Before discussing this, let’s do another run of 13 points (figure 9), setting the reference signal to -25 units and continuing without initialization (the slash command, /):

:\text{R}=-25,/

It is clear that the perceptual signal comes to a steady state quite close to the magnitude of the reference signal, whatever the reference signal may be. The question is, how critically does this tracking effect depend on the input sensitivity and error sensitivity?

Let’s leave the reference signal at -25 and do a run in which the error sensitivity is doubled at the start, and the input sensitivity is doubled halfway through the run. We will start from the previous conditions. The loop gain will now be -40 instead of -10.

:\text{K}2=40,\text{L}=8,/,\text{K}1=2,/

To insure that everything is working correctly, let’s flip the reference signal to +25 units (figure 10):

:\text{L}=16,\text{R}=25,/

While there is an effect on the way the tracking takes place, the only effect of these rather drastic changes in input and error sensitivity is to make the tracking a little better. What about a decrease in these parameters?

:\text{L}=16,\text{K}1=0.5,\text{K}2=10,/,\text{R}=25,/

(Loop gain now 2.5)

Figure 11 shows that the approximation $P=R$ isn’t very accurate any more. For loop gains smaller in magnitude than about 10 (negative), the approximation begins to lose accuracy.

You will notice that doubling the error sensitivity, which doubles the amount of output generated by a given error, does \textit{not} double the amount of output that actually occurs. Far from it. When, for any reason, the loop gain goes up, the steady state error simply gets smaller, assuming that the system remains stable. This fact does violence to the popular idea that the brain commands muscles to produce behavior. If that were the case, doubling the sensitivity of a muscle to the nerve signals reaching it ought to produce twice as much muscle tension. Nothing of the sort happens, unless you’ve lopped off the rest of the nervous system, particularly the feedback paths.

---

\text{Figure 9: The values of variables are listed in this plot. The disturbance value is changed from +25 to -25.}

\text{Figure 10: Change of gain during plot. After 8th line, gain goes from 20 to 40. Reference signal is changed to check operation.}
As long as the loop gain is sufficiently large and negative (−10 or more negative will do for a number), a stable control system will match its perceptual signal nearly to its reference signal, regardless of the reference setting. We are ignoring, of course, transient effects.

All of this was done with the disturbance set to zero. Now let us set the reference signal to zero, and check the second fundamental rule of thumb.

Rule 2: \( (\text{delta } 0) = -(\text{delta } D) \)

This rule requires some interpretation. It says, for the sake of brevity, that (with the reference signal constant) a change in the output quantity is equal and opposite to (the minus sign) a change in the disturbing quantity. Generally, the input and disturbing quantities will affect the input quantity through different physical paths. In our model, the output quantity acts through a multiplier of 0.5, and the disturbance through a multiplier of 0.8. The rule has to be interpreted to mean that the effects of the changes on the input quantity are equal and opposite. We will see this demonstrated.

We will now plot the output quantity, \( O \), the disturbing quantity, \( D \), and the input quantity, \( I \) (to make the above clear). The reference signal could be left where it is, but to avoid confusion let’s set it to zero for this set of plots. The loop gain is set to −10.

: PLOT OID, R=0,K1=1,K2=20,L=1,D=0,
.: L=16,K1=0,5,K2=10,.R=25,/

Let this plot run out, then:
:: D=−15,/

There is some lurching back and forth in figure 12, but in the steady state the behavior of the input quantity shows that the effect of the disturbance is essentially cancelled by the final effect of the output quantity.

If you did some measuring on the plot, you would find that the final value of the output quantity is very close to 8/5 of the value of the disturbing quantity. This follows from three facts: the input quantity ends up nearly at zero; one unit of output has 0.5 unit of effect on the input quantity; one unit of disturbance has 0.8 unit of effect on the input quantity. This is the kind of reasoning that helps in understanding how a control system works.

The primary observation about a control system is always the existence of an input quantity which is stabilized against disturbances by variations in the output quantity. If the input quantity is held essentially...
constant (in the steady state), then one can deduce the relationship between disturbances and the system's output quantity simply from observing the properties of the system's environment. On inspection, an external observer can see both the feedback function and the disturbance function, here multipliers of 0.5 and 0.8 respectively. For any given disturbance, the effect on the input quantity for a constant output quantity can be calculated on purely physical grounds. Since the input quantity remains undisturbed in the steady state, one can then look at the connection between the output quantity and the input quantity, and deduce how the output quantity must change to account for the fact that the input quantity doesn't change.

Thus, in order to predict how this system will react to any external disturbance, it is necessary only to know that the system is a control system and to look closely at its environment. The kind and amount of reaction follow from the nature of the feedback and disturbance functions which are properties of the visible environment.

Most important, as far as the life sciences are concerned, the form and amount of reaction do not depend on any property of the control system; not enough to make any difference. Therefore, when you apply a stimulus and see a response, you are using the organism as a complicated analogue computer in order to study the physics of the local environment. This is not what the life sciences have thought they were doing.

All that remains to wrap up this section is to see the effects of disturbances when the reference signal is set to different values. This will lead to the definition of a useful technical term: the reference level of the input quantity (see figure 13):

If you have a 16 line video display this will scroll past you, losing the early parts, but no matter. The first event is that the reference signal is set to 12, and the input quantity moves essentially to +12. The output quantity goes to +24 in order to accomplish this. Then the disturbing quantity goes to +15, which has the exact effect on the input quantity that +24 units of output quantity have. As a result, the output quantity drops to zero — exactly zero, if you look at the numbers.

In effect, the disturbance, by itself, has enough effect to make the perceptual signal match the reference signal. Looking at figure 8, you can see that this would mean a zero error signal and no drive to the output function. So, whenever the output drops to zero, we know that the perceptual signal is matching the reference signal, even if we can’t see it.

In our model right now, the input sensitivity is 1, so the perceptual signal is numerically equal to the input quantity. That’s a coincidence, since the units are different: physical units outside, impulses per second inside. Even if $K_1$ wasn’t 1, the output would still drop to zero when $P = R$. Thus, we can give a special name to the particular value of input quantity (however created) that brings the error signal, and hence the output quantity, to zero: the reference level of the input quantity. The reference signal clearly determines what this reference level will be, but so does the form of the input function.

**Main Points Reviewed**

All of this is supposed to have established two principal ideas. The first is that control systems control what they sense, not what they do. The second is that control systems act on the outside world only in order to protect a controlled perception against disturbance.

As we have demonstrated these principles, we have established some other odd facts. We have found that the main effect...
If you’re still looking for software for your micro-
I’ve found that
Mad Hatter Software
HAS IT ALL.

**PROGRAM**  
**DESCRIPTION**

**GALACTIC BLOCKADE RUNNER**  
AN EXCITING SPACE WAR GAME WITH GRAPHICS

**SCI-FI GAME SAMPLER**  
3 GAMES—LUNAR LANDER—STAR MONSTER—SPACE BATTLE

**R/T LUNAR LANDER**  
A REAL TIME LUNAR LANDER WITH GRAPHICS

**MICRO-TEXT EDITOR**  
FORMAT TEXT—SAVE & LOAD TO TAPE—OUTPUT TO PRINTER

**OTHELLO III**  
A STRATEGY BOARD GAME—PLAY AGAINST COMPUTER OR OTHERS

**AIR RAID**  
PLAY CHESS WITH YOUR COMPUTER—VARIOUS LEVELS OF DIFF.

**MICRO-CHESS**  
DON'T WAIT FOR OTHERS TO PLAY—YOUR COMPUTER'S READY

**BRIDGE CHALLENGER**  
BLACKJACK WITH HIRES GRAPHICS

**APPLE 21**  
SCI-FI GAMES FOR THE APPLE

**MUSIC MASTER**  
PUT SYSTEM TAPES ON DISK EVEN IF IN SAME MEM AS DOS

**DISK MUSIC MASTER**  
ENTER SHEET MUSIC—THE TRS-80 THEN COMPILES & PLAYS IT

**DISK NUMBER**  
RENUMBER YOUR BASIC PROGRAMS—RENUMBERS EVERYTHING

**DISK RENUMBER**  
SAME AS ABOVE, BUT ON DISK

**PILOT 2.0**  
THE EDUCATIONAL LANGUAGE—IN MACH. LANG.—INC. EDITOR

**PILOT 3.0**  
THE DISK VERSION OF THE ABOVE

**APPLE TALKER**  
YOUR APPLE SPEAKS! NO NEW HARDWARE REQUIRED

**APPLE LIST’NER**  
SPEECH RECOGNITION THE EASY WAY—GREAT WITH THE TALKER

**TIC-TAC-TALKER**  
TIC-TAC-TOE USING SPEECH SYNTHESIS AND RECOGNITION

**SYSCOP**  
MAKE BACKUP TRS-80 SYSTEM TAPES THE EASY WAY

**ANDROID NIM-2**  
GAME OF NIM WITH ANIMATED ROBOTS AND SOUND

**SNAKE EGG**  
A BETTING GAME WITH ANIMATED SNAKES AND SOUND

**LIFE 2**  
100 GEN. PER MIN. LIFE & BATT F OF LIFE W/ANIMATION & SOUND

**DCV-1**  
PUT SYSTEM TAPES ON DISK EVEN IF IN SAME MEM AS DOS

**MUSIC MASTER**  
ENTER SHEET MUSIC—THE TRS-80 THEN COMPILES & PLAYS IT

**DISK MUSIC MASTER**  
SAME AS ABOVE BUT ON DISK W/MANY SELECTIONS

**OPEN'S UP THE WHOLE WORLD OF CP/M SOFTWARE TO THE TRS-80**

---

**Business & Application Software Available**

- Client Billing
- A/R
- A/P
- Gen. Ledger
- Inventory
- Payroll
- Mailing List
- File Handling

For TRS-80 APPLE & OTHERS

Call or Write for Details

---

**Availability from These Fine Micro Computer Dealers**

- Capitol Computer Systems
- Computer & Allied Vision
- Hobby World Electronics
- MJ Electronics LTD

- Computer Village
- Buxhill on Sea, E. Sussex, Eng.

- Computer & Allied Vision
- Buxhill on Sea, E. Sussex, Eng.

- Hobby World Electronics
- 1715 Hill Road, Westboro MA 01611

- MJ Electronics LTD
- 28 Collings Rd.

---

**10% OFF IF YOU ORDER 3 SOFTWARE PACKAGES OR MORE. SEND FOR FREE CATALOG—GIVE TYPE OF COMPUTER**

TO ORDER BY PHONE OR FOR DEALER INFO—CALL—(617) 682-8131
ADD 75c SHIPPING & HANDLING • MASS. RESIDENTS ADD 5% SALES TAX

MAD HATTER SOFTWARE • 900b SALEM RD • DRACUT, MA 01826

---

Circle 202 on inquiry card.
of negative feedback in a control loop is to diminish the effects which disturbances would otherwise have on the system's input quantity. While we have had only one disturbance at our disposal, it should be clear that the number or the causes of disturbances make no difference. If ten different disturbances were acting at once, they could only end up increasing or decreasing the value of the controlled input quantity. Since the system maintains control by acting directly on the input quantity, and not by acting to oppose the cause of the disturbance, the system does not have to take account of the number of causes acting, or the phenomena that are involved. It acts to oppose the net effect of any disturbances on the input quantity.

From the point of view of the behaving system itself, reality consists of the magnitude of one perceptual signal, because that is the only internal representation of the outside world. If the system can be said to have a purpose or intention, it must be to maintain the perceptual signal matching the reference signal. The reference signal specifies to the system what it is to sense, but not what it is to do. The output that matches perceptual and reference signals is determined by the nature of the feedback function and by the strength and direction of any disturbances that may be acting. Whatever sets the reference signal, thus effectively controlling the perceptions of this system, does not have to know anything about how the control system comes up with a matching perception.

What is perhaps most amazing to a person who has not previously worked with negative feedback systems is the capability that this system has to maintain quite precise control over its own perceptual signal, even if its own properties change. If its output apparatus becomes stronger or weaker, or its perceptual apparatus becomes more or less sensitive, there is scarcely any effect on the perceptual signal. As long as some minimum loop gain is maintained and the system does not become unstable and begin oscillating, it does not really matter how much loop gain there is, or whether most of it is in the output or the input function.

A servomechanism engineer might find this approach somewhat odd. Why all this fuss about the system's internal perceptual signal? When you build a control system for a practical use, you worry more about the external variables than internal variables, because the customer is interested in the external variables.

This is exactly the point. Living control systems are not interested in the external variables. They can't be. They don't know about them, except indirectly. All they know is what happens to themselves. The point of behavior is not to accomplish something for a user in the external world, but to affect the system itself. Everything that a living system knows about the outside world has to first exist in the form of perceptual signals, or some other internal effect of external events (not all organisms have nervous systems).

In part 3 we will start looking at living systems more directly, and this will become much clearer. We now know that control systems control, above all, their own internal perceptual signals. Next time we will see why they do that.

In the meantime you might enjoy using this simulator to do further explorations. We have looked into only a few of the questions that might be raised about control systems. The simulator can reveal far more than we have seen. For example, it is instructive to look at the effects of the disturbance strictly from the external point of view (plotting I, O, and D), and then to look at exactly the same effects from inside (plotting P, E, and R). We haven't even raised the question of what a control system looks like when it becomes unstable, how the slowing factors interact with loop gain to determine stability, or what happens when the input function, the output function, or both are nonlinear. Speaking of nonlinearity, you might try rewriting the definition of the feedback function as follows:

\[ 45 \text{DEF FNF}(X) = X \times X \times X / 2048 + X / 2 \]

and then performing some of the experiments again. Try to make the input function logarithmic (adding a constant to make sure you don't make the perceptual signal negatively infinite), and see how the input quantity and perceptual signal behave as the reference signal or disturbance is changed.

The main objective before the next article in this series appears is to understand how a control system controls its perceptual signal, and why an external observer, who doesn't know about the controlled input quantity, might think the disturbance acts on the system to make it respond, like a doorbell. The simulator is there to help you grasp this closed loop phenomenon. I hope it does help.■
CBASIC 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. Each book includes source listings, program and system documentation and user's manual.

What systems are we selling?

* PAYROLL WITH COST ACCOUNTING

* ACCOUNTS PAYABLE AND ACCOUNTS RECEIVABLE

* GENERAL LEDGER

The books sell for $15.00 each. All three are available right now with listings in Wang BASIC. New editions of each book, with listings in CBASIC-2, will be published during the summer of 1979. These new editions will run on many floppy disk-based microcomputer systems.

Don't know programming? Don't want to key in thousands of lines of source code? We have over 130 dealers who will sell you the CBASIC-2 programs on ready-to-run floppy disks. Contact us for the name of your nearest dealer.

CBASIC or Wang BASIC won't work? Take a look at the list of consultants who have converted O&A programs to run on many popular systems, and are waiting to hear from you.

CONSULTANTS...
COMPUTER STORES...
SOFTWARE HOUSES

GOOD NEWS

**CP/M CBASIC***

You can be a dealer for the magnetic surface of O&A's CBASIC-2 software, which runs under CP/M on many floppy disk-based computer systems. We will sell the CP/M magnetic surface only to consultants, computer stores and software houses. Osborne & Associates prefers to write and sell books.

What systems are we selling?

- Payroll with Cost Accounting
- Accounts Payable and Accounts Receivable
- General Ledger

The books sell for $15.00 each. All three are available right now with listings in Wang BASIC. New editions of each book, with listings in CBASIC-2, will be published during the summer of 1979. These new editions will run on many floppy disk-based microcomputer systems.

Don't know programming? Don't want to key in thousands of lines of source code? We have over 130 dealers who will sell you the CBASIC-2 programs on ready-to-run floppy disks. Contact us for the name of your nearest dealer.

CBASIC or Wang BASIC won't work? Take a look at the list of consultants who have converted O&A programs to run on many popular systems, and are waiting to hear from you.

CONSULTANTS...

COMPUTER STORES...
SOFTWARE HOUSES

GOOD NEWS

**CP/M CBASIC***

You can be a dealer for the magnetic surface of O&A's CBASIC-2 software, which runs under CP/M on many floppy disk-based computer systems. We will sell the CP/M magnetic surface only to consultants, computer stores and software houses. Osborne & Associates prefers to write and sell books.

What systems are we selling?

- Payroll with Cost Accounting
- Accounts Payable and Accounts Receivable
- General Ledger

The books sell for $15.00 each. All three are available right now with listings in Wang BASIC. New editions of each book, with listings in CBASIC-2, will be published during the summer of 1979. These new editions will run on many floppy disk-based microcomputer systems.

Don't know programming? Don't want to key in thousands of lines of source code? We have over 130 dealers who will sell you the CBASIC-2 programs on ready-to-run floppy disks. Contact us for the name of your nearest dealer.

CBASIC or Wang BASIC won't work? Take a look at the list of consultants who have converted O&A programs to run on many popular systems, and are waiting to hear from you.
AMRAD members. Contact Amateur publication which is mailed to all each month at 8 PM at the Patrick Henry dissemi­nate technical information; and applied research; or­ganize forums and vocate design of experimental equip­ment and techniques; promote basic and applied research; organize forums and technical symposiums; collect and disseminate technical information; and provide experimental repeaters.

Meetings are on the first Monday of each month at 8 PM at the Patrick Henry Branch Library, 101 Maple Av E, Vienna VA. The AMRAD Newsletter is a monthly publication which is mailed to all AMRAD members. Contact Amateur Radio Research and Development Corp, 1524 Springvale Av, McLean VA 22101.

The Amateur Radio Research and Development Corp (AMRAD) is a Virginia based club of over 200 radio and computer amateurs. The purpose of the club is to develop skills and knowledge in radio and electronic technology; advocate design of experimental equipment and techniques; promote basic and applied research; organize forums and technical symposiums; collect and disseminate technical information; and provide experimental repeaters.

Meetings are on the first Monday of each month at 8 PM at the Patrick Henry Branch Library, 101 Maple Av E, Vienna VA. The AMRAD Newsletter is a monthly publication which is mailed to all AMRAD members. Contact Amateur Radio Research and Development Corp, 1524 Springvale Av, McLean VA 22101.

Sacramento Microcomputer Users Group

The Sacramento Microcomputer Users Group newsletter, Push & Pop, comes our way on a monthly basis. The most recent contained articles on CP/M Sleuth and CP/M to Pencil and Pencil to CP/M in Z-BD. The general club meetings are the fourth Tuesday of every month at 7:30 PM. The club also spon­sors a P&T workshop, SOL workshop, and hardware study group. For meeting location and more information about this group contact SMUG, POB 161513, Sacramento CA 95816.

Attention: Fox Valley Area Illinois Computer Enthusiasts

A fairly new computer club is getting started in the Fox Valley area of Illinois. Called the Microprocessor User Group, they are primarily an Apple user group but wish to include all processors in their discussions. With a current membership of 50, it is the club's intention to publish a newsletter and a calendar. Meetings are held in the Fermi National Accelerator Laboratory Hi-Rise on the main floor in the SW meeting facility on the third Monday of every month at 8 PM. Contact Mike Urso, 641 Woodlawn, Aurora IL 60506.

Pacifica TRS-80 Club

The Pacifica TRS-80 Users Group is meeting in the Radio Shack store in the Eureka Square Shopping Center which is located about 10 miles from San Francisco. They meet the second and fourth Thursday of the month to exchange programs and ideas regarding the TRS-80. All individuals are cordially invited to attend one of their meetings. Contact John F Strazzarino, 637 Brussels St, San Fran­cisco CA 94134.

Evansville Indiana Computer Club

Evansville IN does have a computer club according to a recent letter from Robert Heerdink. He is concerned with increasing the club's membership and encourages any interested individuals in that area to attend one of their monthly meetings. The group is interested in several types of microcom­puters. They usually meet at the Blind Association, Second Av and Virginia, at 7:30 PM on the second Wednesday of the month, unless he recommends checking with him first. Robert can be reached at the Evansville Computer Club, c/o National Sharedata Corp, POB 3895, Evansville IN 47737, (812) 426-2725.

Theater Computer Users Group

The Theater Computer Users Group is sponsored by Theater Sources Inc, a corporation created to gather and distribute information about theater in the United States. Their newsletter, called TCUG Notes, is $4 per year, and back issues are available. Contact TCUG Notes, Mike Firth, Editor, 104 N St Mary, Dallas TX 75214.

Alamo Computer Enthusiasts

Located in San Antonio, the Alamo Computer Enthusiasts meet on the third Friday of each month at the Norris Technical Center, Room 208, St Philip's College, San Antonio TX. Their monthly newsletter is available for a one year subscription fee of $2. Contact Dave Fashenpour, 5411 Cerro Vista, San An­tonio TX 78233.

LISP Users Newsletter

The first issue of the Lisp Users Newsletter is an announcement of this new organization which is designed to
spread information about applications, implementation and general information on LISP-like languages. It will pay particular attention to those LISPish things which are within the realm of currently available microprocessor systems. As an initialization process, a mailing list of prospective interested individuals is being compiled. If you are interested, send your name and address, background, interest in and knowledge of LISP, what system (if any) you own, and memory capacity (internal and mass storage) to John R Allen, 1821 5 Bayview Dr, Los Gatos CA 95030. Please include any suggestions you may have about organization or topics you would like to see discussed. John requests that you submit a dollar to cover postage and duplication of the newsletter.

Free TRS-80 Newsletter

The free TRS-80 Bulletin, which prints club and product news about the TRS-80, is sporting a new format and is now monthly. The Bulletin is an offshoot of TRS-80 Computing (see BYTE Clubs and Newsletters December 1978). For a free copy of the Bulletin write to the Computer Information Exchange Inc, POB 158, San Luis Rey CA 92068.

Ventura County TRS-80 Club

This group is comprised of about fifty computer enthusiasts living in the Ventura County area who could lend their expertise to the creation, organization and realization of this idea. Interested persons should contact him by mail, in care of the station or by phone at (417) 271-1036.

Chess Tournament

The second London microprocessor chess tournament will be held in the West Centre Hotel, Lilac Road, Fulham, London England, from November 1 thru 2 1979. Any interested company wishing further details should write to David Levy, c/o Personal Computer World, 62a Westbourne Grove, London, W2.

This year's event will be the first European Open Microprocessor championship. The highest placed participants will automatically qualify for places in the final of the first World Micro Championship which is scheduled to be held in London in 1980.

I.E.C. '80 Issues Call for Papers

I.E.C. '80, the Sixth Annual Conference and Exhibit on Industrial and Control Applications of Microprocessors, will be held at the Sheraton Hotel in Philadelphia next March 17 thru 19, 1980. I.E.C. '80 will offer an exhibition and technical program dealing with the current and new work of industrial microprocessor applications. Papers dealing with such areas as automotive diagnosis and operation, intelligent instrumentation, transducers and sensors, automated manufacturing, numerical control and robotics, and biomedical control and monitoring are invited. Ten copies of the paper in extended summary form to 600 words long and an abstract of no more than 40 words describing work not generally published or previously presented, should be mailed by September 14 1979 to H T Nagle Jr, Electrical Engineering, Auburn University, Auburn AL 36849. The extended summary will be used for paper selection and session assignment and should clearly define the salient concepts and novel features of the work described. Notification of acceptance and format required for publication in the I.E.C. '80 proceedings will be sent by October 19 1979.

Forth Inc Announces 13 Software Seminars

Forth Inc, producer of Forth based software systems for micro and mini computers, has announced a schedule of 13 seminars throughout the US this summer. The seminars, which are half day events, cover design, application and implementation of the news letter.
Byte News Update

Byte News in the March 1979 issue began with a paragraph listing computer manufacturers that support Pascal. For a year, Texas Instruments has supported Pascal which can be implemented on the Model 990 minicomputer in the form of T1 Pascal (TIP). There are also two executive operating systems available which are appropriate for use with Pascal.

The Texas Instruments Microprocessor Executive Library (TIPMX) is a collection of operating system components available to users of the TMS 9900 family of microprocessors. Minimum development tools required for use with TIPMX are a text editor, an assembler and a link editor, all of which are available with the purchase of a Texas Instruments software license. TIPMX may be used on either the floppy disk based FS990 development system, or any of the several hard disk systems such as the DS 990. TIP and its extensive run time support library are available on the hard disk based system and are highly recommended for use with TIPMX.

The latest addition to TI's software support is called Texas Instruments' Modular-Based Executive in Read only memory (TIMBER). TIMBER is targeted for users of the TM 990 series of microcomputer modules. In this way it is possible to do Pascal software development on a low cost system incorporating the TM 990/302 software development system. The fundamental benefit of TIMBER is that it is pre-packaged in erasable read only memory, thereby freeing the user from the concern of inadvertently destroying the operating system.

In the second paragraph of Byte News, Intel was mistakenly credited with being the first to market a 32 K byte programmable read only memory. Texas Instruments introduced the world's first 32 K programmable read only memory in April of 1978. The TMS 2512 is organized as 4 K by 8 bits and operates from a single +5 V power supply with standby power dissipation of 50 mW typical. A +25 V supply is only needed for programming.

Dale R Gibble
Project Engineer Texas Instruments
POB 1443 MS 6750
Houston TX 77001

BYTE's Bugs

A Reorganized Wine Cellar

As one who enjoys both wine and computers, I must comment on several inaccuracies in the article entitled "Computerized Wine Cellars" (February 1979 BYTE, page 128). The second wine in listing 1 is Raauenthal Rothenberg Riesling Spatlese 1976 — von Simmern. The varietal name is not, as listed, Rhine, which is a geographical area, but Riesling, which is the name of the best grape variety produced in Germany. The Rheingau, which is the Rhine sub-area from which this wine comes, produces Riesling almost exclusively, but other Rhine areas produce Sylvaner, Müller-Thurgau, and other varietals as well.

More strangely, the producer is listed as Rauenthaler. In top quality German wines, the first name refers to the village and the second to a particular vineyard. Here, we have the village of Rauenthal and the vineyard Rothenberg. The producer is the Graf (Count) von Simmern.

In any case, I have recently enjoyed tasting this wine and I can verify the accuracy of the tasting notes in the listing — it is a super wine, and in wine drinking, that's what counts!

Larry Rosenblum
12008 Trailridge Dr
Potomac MD 20854
THE LEADER IN MICROCOMPUTER EDUCATION PRESENTS

BEST-SELLING BOOKS
Used by Universities and Industry worldwide (ten languages)

C200 - AN INTRODUCTION TO PERSONAL AND BUSINESS COMPUTING
Rodnay Zaks, 250 pp
A comprehensive introduction to small computers, their peripherals, and what to select. $4.95

C201 - MICROPROCESSORS: FROM CHIPS TO SYSTEMS
Rodnay Zaks, 416 pp
The basic text on all aspects of microprocessors and the assembly of a system. $9.95

C202 - PROGRAMMING THE 6502
Rodnay Zaks, 320 pp
A complete introductory programming text for the 6502. $10.95

D302 - 6502 APPLICATIONS BOOK
Rodnay Zaks, 286 pp
Connecting a 6502 board to the outside world; from home alarm to music and industrial control. $12.95

C280 - PROGRAMMING THE Z80
Same as C202 for the Z80. $10.95

AVAILABLE AT BOOKSTORES, COMPUTER STORES, AND ELECTRONIC SHOPS EVERYWHERE!

SELF-STUDY COURSES ON CASSETTES
“The time-efficient way to learn”™

Self-study courses include two to eight audio-cassettes and a special book. They offer the fastest way to learn the topic covered (from 1/2 to two days). Highly effective.

1 - INTRODUCTORY - SHORT (2.5 hrs ea)
$1 - INTRODUCTION TO MICROPROCESSORS $29.95
$2 - PROGRAMMING MICROPROCESSORS $29.95
$3 - DESIGNING A MICROPROCESSOR SYSTEM $29.95
$10B - INTRODUCTION TO PERSONAL AND BUSINESS COMPUTING $21.90

2 - INTRODUCTORY - COMPREHENSIVE (10 to 12 hrs ea)
$81 - MICROPROCESSORS $59.95
$82 - MICROCOMPUTER PROGRAMMING $59.95

3 - SPECIALIZED (4.5 to 6 hrs ea)
$83 - MILITARY MICROPROCESSOR SYSTEMS $49.95
$85 - BIT-SLICE $49.95
$86 - INDUSTRIAL MICROPROCESSOR SYSTEMS $49.95
$87 - MICROPROCESSOR INTERFACING $49.95

TO ORDER

By Phone: 415 848-8233, Visa, M.C., Amer Express.
By mail: Include payment.
Shipping: add $1.50 per book (UPS) or 75¢ 14th class - allow 4 weeks.
Double for cassettes and overseas.
Tax: in California add tax.
FREE DETAILED CATALOGUE

2020 Milvia Street
Berkeley, CA 94704
Tel 415 848-8233 Telex 336 311

Circle 356 on inquiry card.
Creativity in Computer Music

The most important result of computer applications in music is that they have encouraged a variety of new ideas to be formulated, tested, and reformulated in a short period of time. Concepts that previously would have taken much longer to develop and verify have already been brought forth. Musical thought is consequently on a higher level today. This applies to all aspects of work with computers—a much to research as to creative work, but perhaps less to music than to some other fields. The increasing availability of microcomputers can only further this trend.

The distinction between research and creative work is artificial, for much research is creative—certainly when it pursues new ideas. Computer work in music has for many years been carried out in the creative areas of musical composition and sound synthesis, and the research areas of musical analysis and music theory. In this discussion I will not consider such areas as music bibliography, music printing, CAI (computer-assisted instruction), and related disciplines. While much important work has been done in these areas, they are not generally concerned with creative or conceptual problems. Music bibliography essentially involves an information storage and retrieval system. Music printing is a problem of automation. CAI certainly encompasses creative work, but thus far only the most rudimentary instructional tasks have been delegated to computers, and then only with limited success.

Instead, I will discuss music theory, analysis, sound generation, and composition and their relation to the computer. While we are discussing the question of musical tasks in general, it is relevant to point out that any musical task is in principle subject to delegation to a computer. Musical tasks include any activities carried out by human beings that involve interaction with music.

Music is composed by composers in ways that, traditionally at least, involve only their own internal ideas, realized sometimes with the assistance of an instrument like the piano. Music is performed by performers who usually play from printed notation. Music is heard by listeners who may then engage in reflection, evaluation, or criticism. Musical documents such as recordings or scores are kept in libraries where they are filed according to careful systems of classification. Computer composition, performance, printing, recording, documentation, bibliography—even evaluation or criticism—all are feasible.

The problems involve simulating the sensory processes with which people interact with music—listening, writing, visual communication—and the mental processes with which they make their conclusions about music. The sensory processes are simulated by the I/O (input/output) devices on computers, and since computers cannot do all of these tasks in the same way that people can, various languages have been devised to enable translation from the human form into a form computers can read. The mental processes are simulated by the programs.

Now that we have described these general features of computer work in music, let us consider some specific problems concerning the four disciplines mentioned above and how using a computer has encouraged the formulation of new concepts in creative ways.

Music Theory

There has always been a close relationship between the discipline of music theory and other fields, particularly analysis and composition. In the twentieth century, composers have produced some highly original theories that they have attempted to apply to their compositions, either before or after they have written them. The term "speculative music theory" describes theories conceived before the music illustrating them is written.

After the music has been composed, it is usually a simple matter to verify whether
At last. A controller you'll never have to throw off the bus.

If you've got a floppy disk controller on your S-100 bus, you've got a big problem. Because when you want to upgrade your system from single to double density, or from single-sided to double-sided, or from 5" to 8" floppy, you'll have to throw that old controller off the bus.

But not any more you don't. Because at DATASPEED, we've developed one controller that works with all four kinds of floppy disk drives. And doubles the density, too.

Works with all floppy disk drives.

Single-sided minifloppys. Double-sided minis. Single-sided maxis. Even the new double-sided maxis with up to 8 megabytes. This is the only controller in the world that works with all four.

Handles any upgrade.

When you want to go from single-sided to double-sided drives, this great new controller reads each side automatically. And you can upgrade from 5" to 8" disks just by changing a simple Header socket.

Packs in twice the data.

Now you can pack double density data on every side of every disk. Because we've designed in a bootstrap PROM that automatically controls any shift from single to double density. And, of course, when you're running double density, you can transfer data in half the time.

So when you buy this controller, and a compatible CP/M, we're including a free software package that lets you transfer all your files to double density.

We call it The Conductor.

The DATASPEED Floppy Disk Controller. It handles all your upgrades. It packs in twice as much. And at only $295, it's even the cheapest way to go. That's why this is one controller you'll never have to throw off the bus. And that's why we had to call it The Conductor. Ask for it at your local computer shop or send $295 for immediate delivery. By the way, we have Double-Density CP/M for only $100. Complete disk drive systems are available.

Write us for more information. Dealer inquiries welcome. And watch our future ads for trade-in offers. DATASPEED, INC., 1302 Noe Street, San Francisco, CA 94131.

THE CONDUCTOR™

Minifloppy is a TM of Shugart Associates. CP/M is a TM of Digital Research.
a theory applies to it, assuming that the theory is adequately formulated. For example, Schoenberg's conception of atonal music and later his "method of composition with 12 tones" originated as speculative ideas which were described, albeit confusingly, in his books *Harmonielehre* and *Style and Idea*. More recent composer/theorists like Milton Babbitt have extended Schoenberg's original ideas to even greater levels of abstraction.

The availability of computers has spurred great efforts to test these theories and other ideas to a greater degree than people had been willing to attempt before. (Though most theoretical work has been done on large general-purpose computers, much can now be done on microcomputers, as is shown in our example of pitch structures.) Concomitantly, the results obtained by these researchers have brought forth new ideas and exposed the inadequacies in old ones.

Part of the reason for such success has been the fact that it is by no means obvious or easy to explain how theoretical ideas apply to a piece of music in detail, and people have had to clarify their methods in ways that go beyond what had been accepted before in noncomputer work. Computers do not presently have the ability to *listen* (in the sense that a person "listens") to a piece of music to decide whether or not a given idea applies to it. Researchers have instead formulated methods that involve detailed inspection of the *notation* for a piece. Sometimes these produce results that seem mystifying or incorrect when compared with results reported by listeners. This conflict demonstrates the difficulty of accepting unintuitive ideas about music and the question of whether or how listeners ought to revise their listening habits in order to perceive musical structures that are verified to exist in the music.

Theories, in the sense employed thus far, are ideas that explain the structure of entire compositions or passages of music. Few musical theories, even that of triadic tonality, are in fact expressed in any such detail. Usually there is a large gap between the theoretical constructions and their applications to the music that is supposed to be filled by the reader. This is obviously true of such modern concepts as pantonality (see reference 6), or of attempts to perceive Schoenberg's 12 tone compositions in a certain key (see page 407 of reference 7). Allen Forte's theory of set complexes (reference 8) presents a collection of ideas that would still not be a theory in the sense described above, but he nevertheless has much more success...
LET'S GET PERSONAL
How to use your personal microcomputer to its fullest potential

PIMS — PERSONAL INFORMATION MANAGEMENT SYSTEM
By Madan Gupta

- This book describes a data base management system designed for personal use on small computer systems using BASIC. Along with complete source listings of the program and comprehensive operating instructions, the book discusses the personal microcomputer and its potential for personal use. The emphasis is on un"
- This book describes a data base management system designed for personal use on small computer systems using BASIC. Along with complete source listings of the program and comprehensive operating instructions, the book discusses the personal microcomputer and its potential for personal use. The emphasis is on un"
- Improve your style with BITS Programmer's Pads.™ One side is an ingenious memory/register allocation form, custom-tailored to the microprocessor you're programming: 6800A, 6800, 6502, Z-80, or 1802. The other side is for your program, with pre-numbered lines and columns for addresses, object code, labels, instructions, and those essential comments. Write your source code first and later assemble by filling in the object codes and assigning memory addresses. (If you're hand assembling your programs, these pads are a must!) 50 page pads on durable paper. Pre-punched for a 3-ring notebook, too.

YOUR HOME COMPUTER
By James White

- A clearly written, nontechnical description of personal computers that requires no prior knowledge of computers or electronics. The emphasis is on understanding; over 100 illustrations are included. Topics include: computing and you; communication inside a computer; computer thought processes; fixed memory; inputs and outputs; peripherals; systems components; how to choose a microcomputer; and so on. Your Home Computer is the ideal book for readers who thought they could never understand how computers work. The best part is that this is both easy and fun to read.

32 BASIC PROGRAMS FOR THE PET COMPUTER
By Tom Rugg and Phil Feldman

- Here are 32 fully documented programs, ready to run on your Commodore PET 2001 computer, including applications programs, educational programs, games, graphical display programs, math programs, and more. The program instructions are detailed enough to allow you to modify any program and sample runs are included. PET users, don't leave your PET on the shelf, put it to work for you!

DIAL YOUR CHARGE CARD FIRE
ORDERS TOLL FREE
800-258-5477
(in NH call 924-3355) Mon-Fri 9-5

For your convenience in ordering, please use this page plus the order form on page 137
in describing how his ideas relate in specific ways to actual musical content and the difficulties encountered in this process.

Forte's theory is a striking example of a series of ideas that grew out of a continuing interaction with computers. In his case, as well as that of this author, computers figured prominently in the development of the concepts employed in formulating the theory. Composers have sometimes referred to these concepts as precompositional ideas.

In my own work (described in "Some Combinational Properties of Pitch Structures," Perspectives of New Music, volume 4, number 1, fall/winter 1965, pages 45 thru 61) the basic concepts were pitch, pitch class, pitch class collection, and pitch structure. A computer was used to generate the basic sets of all of these elements, and then to test various operations such as inversion and cycle of fifths equivalence. The fact that a computer was used in this work meant that my list of pitch structures generated by the computer was one of the first accurate lists ever produced. Howard Hanson's count of distinct pitch sets in The Harmonic Materials of Modern Music: Resources of the Tempered Scale, Irvington Pub, NY, 1961, is incorrect.

Forte's list published in "A Theory of Set Complexes for Music," Journal of Music Theory, volume 8, number 2, winter 1962, is correct according to his definition, but his definition does not reflect current practice. I have continued to use computers in my work, which now includes computer composition as well as theory development.

It is not difficult to see that the continuing use of computers in music theoretical work will have beneficial results, if only because it will force the theorists to formulate their ideas in more specific terms. Such continuing investigations could reveal and clarify many inadequacies in the theory of harmony or tonality as presently stated in harmony textbooks.

In fact, the computerization of the principles stated in any of these books ought to point up all kinds of basic problems that are never acknowledged — for example, just exactly what the theory explains about which music and what that has to do with any activity the reader would be likely to engage in with respect to either the theory or the music. Harmony, as we understand it from harmony texts, is simply the translation of a piece of music into a different notation or vocabulary — a vocabulary that is imprecise and overlooks many of the most important characteristics of the music itself. The reasons why such a translation is made and what advantages it possesses are unclear. Because they have to clarify their purposes to impersonal machines, computer researchers are not able to ignore these questions as easily as traditional ones.

Musical Analysis

Computer work in musical analysis highlights the question: what do we really want to know about a piece of music? There are
New RAM Prices.
From The Dynamic Memory Company.

16K—$249  32K—$375  48K—$500  64K—$625

Ever since we started making these memory boards over a year ago we have continued to lower our prices to stay competitive. Due to your confidence in us, we are again able to lower our prices! Our reliability has been proven by months of superior performance in thousands of installations. Our low-power boards are being used by quality-minded systems manufacturers across the country and overseas.

4MHz boards now available.
After receiving hundreds of requests, our engineering staff has come up with a new version of our board which runs on 4MHz Z-80 systems. It wasn’t easy to come up with a high speed board which would operate as reliably as our 450ns version, but after months of careful design and testing, we did it. The price of the 250ns board is $10 per 16K additional.

All of our features remain.
Our boards didn’t become great sellers only because of the price. We still offer you our deselect feature which allows our RAM to overlap with any fixed memory areas in your system. Also, the RAM area of our board is fully socketed so that you can expand the board yourself.

Other standard features include:
plug selectable addressing on 16K boundaries (shorting plugs are placed over wire-wrap pins to address the board — located on the top of the board for easy changes), S-100 and Z-80 compatibility and totally invisible refresh — no wait states.

Fully assembled, tested, and guaranteed.
All of our boards go through a rigorous testing procedure. They are then placed on burn-in running a series of memory tests to detect any other possible faults. After you receive the board, you are backed by us with a one year warrantee.

Low power consumption keeps your computer from “losing its cool.”
The total power consumption of our 16K board is typically less than 4 watts (+8V @ 300ma, +16V @ 150ma and −16V @ 20ma). Boards with additional memory typically increase power consumption only 1 watt per 16K!

Standard S-100 Interface.
Our board is designed to interface with any standard S-100 CPU. All of the timing of the board is independent of the processor chip, and the board is set up for different processors by changing two plugs on the board.

Contact your local dealer.
To find out more about our RAM boards, contact your local dealer. If he is unable to help you, call or write us for a fast response.
Central Data Corporation, 1207 North Hagan Street, Champaign, IL 61820. (217) 359-8010

Central Data
two ways in which computer programs attempt to answer this question. One is to pose a traditional question to which we supposedly know the answer, in the hope that the task of carrying this out on the computer will improve our understanding; in other words, it helps us understand musical intuition. The second way is to ask the computer to solve problems we have not yet understood when traditional methods have been used. Serious research will always be concerned with questions of this sort, but unless those of the first sort can also be handled we will be in doubt about the success of the second.

There have been many research projects in musical analysis, most of which are ongoing and have thus far produced only tentative or incomplete results. In my opinion, the most important conflict in this area concerns both purpose and procedure: should computers be used to examine the characteristics of a single work, or rather to analyze the common properties of whole bodies of works?

Some of the most significant results in musical analysis have been obtained by theorists who are more interested in the theoretical problems than the musical works, and who have analyzed works more in order to prove their theories than to discover new aspects of the music. I have already mentioned the work of Allen Forte, who has produced excellent analysis of diverse atonal works with the assistance of a computer. Another example is the well-known "Joquin Project" conducted by Professors Arthur Mendel and Lewis Lockwood at Princeton University. While this project is remarkable for the sheer quantity of music that has been encoded for study, all analytical work has been carried out from the standpoint of music analysis according to the theoretical principles formulated by contemporaries of Josquin. (See reference 9. Josquin des Prés, c1445-1521, was a prominent composer of early music.) Few studies have produced results as successful as these. (Some are absurd. See my review of "Two Parameters of Melodic Line as Stylistic Discriminants" by David Sheldon Lewis and "Some Techniques for Computer-Aided Analysis of Musical Scores" by Donald Margedo Pederson in Perspectives of New Music, volume 9, number 2 and volume 10, number 1, 1971.)

One of the most time-consuming aspects of computer analysis of music has been the development of encoding languages in which musical notation can be transcribed for computer input. Analytical studies have become studies of musical notation, not of sound or auditory experience, and are thus open to the objections raised above. Many languages have been developed for different purposes, and many of them choose to represent the notational characteristics in very different ways. There has been much unfruitful debate over these languages prompted by some researchers trying to get others to adopt their language as universal. One researcher, Eric Regener, has instead taken the intelligent line of writing programs to transcribe other notations into his.

An important quality of these languages is that each was developed with a different purpose in mind, and thus each emphasizes different aspects of the music. Some languages developed for the purpose of music printing (like DARMS) are necessarily quite complex in their methods of representing spacing between notes and other features of page layout that are irrelevant for languages concerned with analytical work. Other languages developed for use with a specific body of music (like IML) cannot be used for other types of music because they do not provide methods for encoding features that do not occur in the works for which the languages were developed.

The point I would like to make about these languages is that any notation expresses a conceptual structure of the music
Basic Time
INTRODUCES THE $15,995*
10MB HARD DISK COMPUTER

Complete System Includes:

★ 10 MILLION CHARACTER HARD DISK STORAGE
★ DISK OPERATING SYSTEM & EXTENDED BASIC
★ POWERFUL 16-BIT CENTRAL PROCESSING UNIT
★ VIDEO DISPLAY TERMINAL
★ 150 CPS LINE PRINTER AND PRINTER STAND
★ DESK ENCLOSURE

1215 E. El Segundo Boulevard
El Segundo, California 90245
213 / 322-4435
it represents. Printing languages are very specific about graphical features because those are important aspects of the intended result. Other languages developed for analysis represent elements like pitches and rhythms as the basic aspects to be examined, and provide many ways for representing nuances in these domains but not others. For another example, guitar notation providing only a melodic line and a basic chord pattern is a remarkably efficient means of indicating a diverse quantity of sounds in a few symbols, and the symbols convey a sense about the irrelevance of certain nuances in the chord structure that would be unthinkable in other music. No language currently in existence is universally satisfactory and efficient to represent all music that could be transcribed for any purpose. Arguments about encoding languages show only a lack of understanding of the diverse and even cross-disciplinary purposes for which certain research is carried out.

**Sound Generation**

In the area of sound generation, computers have already had a significant impact on the music of our time, and the future is even more promising. (An excellent description of the facilities needed to produce music on a microcomputer is contained in Tom O'Haver's article "Audio Processing with a Microprocessor" in June 1978 BYTE, page 166.) To explain the reasons for this, it is necessary to review some of the history of both computer generated music and electronic music, of which computer music is a part.

Electronic music has been shaped by several disparate influences. Historically, the first was the use of nonconventional sounds that were previously considered to be nonmusical. Early works were based on the sounds of railroads, water dripping, and other noises of everyday experience and of nature. Another significant influence was the availability of analog electronic sound generating and modifying equipment, which is now packaged into devices called synthesizers. (A third influence, less important than the first two, was the use of speech and language in noncommunicative ways.) Music based on sounds of nature tended to be extremely complex in the foreground, whereas music produced by electronic equipment often lacked dynamic variations in tonal characteristics.

Today, these early tendencies have been mollified as a result of the experiences of many people working with these ideas, often in conjunction with computers. People have begun to analyze the characteristics of nonconventional sounds, often by computer, to discover and generalize the properties of interest in them. Thus, early exploratory use of these sounds has now given way to conceptual thinking about them. Such thinking is a natural result of processes which people must employ to generate sounds by computers, because users must present information to the computer in concrete ways. A computer cannot generate a sound from a person's abstract recollection of what it sounds like.

Sounds produced by electronic music synthesizers have evolved from the life-less organ-like sounds that contain no dynamic variation in tonal characteristics to sounds that mimic live musical instruments. A synthesizer is, indeed, a musical instrument, and it is natural that people would begin to develop performance techniques once they are able to work with them for a while. Such expert performers as Walter Carlos and Isao Tomita can routinely produce any quality such as vibrato, tremolo, or dynamic spectral variations. They have also developed excellent methods of imitating specific instruments, particularly woodwinds, brass, and percussion (solo string sounds are easier to imitate than the lush sounds of an entire string section).

The trouble with this procedure is that even expert synthesizer performers now at least partly judge their work not by the originality of the sounds they produce, but by their resemblance to the familiar sounds of musical instruments. This is in complete contrast to computer music composers.

To cause a computer to generate music, it is necessary for the composer (who is actually the performer in this case) to provide a detailed description of the sound desired. The description can be anything that is mathematically sufficient to cause the desired properties to be produced by the computer. Any describable sound can be produced; the limitation is not in the capability of the computer but rather in the ability of composers to provide adequate descriptions of what they want. A detailed explanation of how computers generate musical sound is contained in Electronic Music Synthesis (see reference 3).

This factor has been one of the primary reasons for the recent interest in the analysis of sounds of all types. Computer music synthesis is thus not limited by the comparison of the results produced to any pre-existing standard. Indeed, composers are encouraged to be creative with the qualities of the sounds they produce, by the very procedure by which they must work.

Of course, people who work with syn-
5 reasons why you should not buy the electric pencil II

Check the appropriate box(es):

☐ You love typing the same copy 20 thousand times a day.
☐ Your secretary can type 250 words per minute.
☐ You're dying to spend $15,000 on a word processing system, just for the tax investment credit.
☐ All your capital assets are tied up in a 10-year supply of correction fluid.
☐ You never commit a single thought to paper.

If you have checked one or more boxes, you do not need The Electric Pencil. On the other hand, you may want to join the thousands of people who haven't checked a single box.

The Electric Pencil II is a Character Oriented Word Processing System. This means that text is entered as a string of continuous characters and is manipulated as such. This allows the user enormous freedom and ease in the movement and handling of text. Since line endings are never delineated, any number of characters, words, lines or paragraphs may be inserted or deleted anywhere in the text. The entirety of the text shifts and opens up or closes as needed in full view of the user. The typing of carriage returns or word hyphenations is not required since lines of text are formatted automatically.

As text is typed and the end of a line is reached, a partially completed word is shifted to the beginning of the following line. Whenever text is inserted or deleted, existing text is pushed down or pulled up in a wrap around fashion. Everything appears on the video display as it occurs, which eliminates guesswork. Text may be reviewed at will by variable speed scrolling both in the forward and reverse directions. By using the search or replace functions, any string of characters may be located and/or replaced with any other string of characters as desired.

Numerous combinations of line length, page length, line spacing and page spacing permit automatic formatting of any form. Character spacing, bold face, multicolumn and bidirectional printing are included in the Diablo versions. Multiple columns with right and left justified margins may be printed in a single pass.

Wide screen video

Versions are available for Imsai VIO video users with the huge 80x24 character screen. These versions put almost twice as many characters on the screen!!

CP/M versions

Digital Research's CP/M, as well as its derivatives, including IMDOS and CDOS, and Helios PTDOS versions are also available. There are several NEC Spinwriter print packages. A utility program that converts The Electric Pencil to CP/M to Pencil files, called CONVERT, is only $35.

Features

- CP/M, IMDOS and HELIOS compatible
- Supports four disk drives
- Dynamic print formatting
- DIABLO and NEC printer packages
- Multi-column formatting in one pass
- Print value chaining
- Page-at-a-time scrolling
- Bidirectional multispeed scrolling controls
- Subsystem with print value scoreboard
- Automatic word and record number tally
- Cassette backup for additional storage
- Full margin control
- End-of-page control
- Non-printing text commenting
- Line and paragraph indentation
- Centering
- Underlining
- Bold face

Upgrading policy

Any version of The Electric Pencil may be upgraded at any time by simply returning the original disk or cassette and the price difference between versions, plus $15 to Michael Shroyer Software. Only the originally purchased cassette or diskette will be accepted for upgrading under this policy.

Have we got a version for you?

The Electric Pencil II operates with any 8080/8086 based microcomputer, which supports a CP/M disk system and uses an Imsai VIO Printer Processor Tech. VDM-1, Polymorphic VT, Solid State Music VB-1B or Vector Graphic video interface. REX versions also available. Specify when using CP/M that has been modified for Micropolis or North Star disk systems as follows: for North star add suffix A to version number; for Micropolis add suffix B, e.g., SS-II A, DV-II B.

<table>
<thead>
<tr>
<th>Version</th>
<th>Video</th>
<th>Printer</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-II</td>
<td>SOL</td>
<td>TTY or similar</td>
<td>$225.</td>
</tr>
<tr>
<td>SP-II</td>
<td>VTI</td>
<td>TTY or similar</td>
<td>$225.</td>
</tr>
<tr>
<td>SV-II</td>
<td>VDM</td>
<td>TTY or similar</td>
<td>$225.</td>
</tr>
<tr>
<td>SR-II</td>
<td>REX</td>
<td>TTY or similar</td>
<td>$250.</td>
</tr>
<tr>
<td>SI-II</td>
<td>VIO</td>
<td>TTY or similar</td>
<td>$250.</td>
</tr>
<tr>
<td>DS-II</td>
<td>SOL</td>
<td>Diablo 1610/20</td>
<td>$275.</td>
</tr>
<tr>
<td>DP-II</td>
<td>VTI</td>
<td>Diablo 1610/20</td>
<td>$275.</td>
</tr>
<tr>
<td>DV-II</td>
<td>VDM</td>
<td>Diablo 1610/20</td>
<td>$275.</td>
</tr>
<tr>
<td>DR-II</td>
<td>REX</td>
<td>Diablo 1610/20</td>
<td>$300.</td>
</tr>
<tr>
<td>DI-II</td>
<td>VIO</td>
<td>Diablo 1610/20</td>
<td>$300.</td>
</tr>
<tr>
<td>NS-II</td>
<td>SOL</td>
<td>NEC Spinwriter</td>
<td>$275.</td>
</tr>
<tr>
<td>NP-II</td>
<td>VTI</td>
<td>NEC Spinwriter</td>
<td>$275.</td>
</tr>
<tr>
<td>NV-II</td>
<td>VDM</td>
<td>NEC Spinwriter</td>
<td>$275.</td>
</tr>
<tr>
<td>NR-II</td>
<td>REX</td>
<td>NEC Spinwriter</td>
<td>$300.</td>
</tr>
<tr>
<td>NI-II</td>
<td>VIO</td>
<td>NEC Spinwriter</td>
<td>$300.</td>
</tr>
<tr>
<td>SSH</td>
<td>SOL</td>
<td>Helios/TTY</td>
<td>$250.</td>
</tr>
<tr>
<td>DSH</td>
<td>SOL</td>
<td>Helios/Diablo</td>
<td>$300.</td>
</tr>
</tbody>
</table>

Attention: TRS-80 Users!

The Electric Pencil II has been designed to work with both Level I (16K system) and Level II models of the TRS-80, and with virtually any printer you choose. Two versions, one for use with cassette, and one for use with disk, are available on cassette. The TRS-80 disk version is easily transferred to disk and is fully interactive with the READ, WRITE, DIR, and KILL routines of TRS-DOS 2.1.

Demand a demo from your dealer!
that synthesizers are not limited either; nevertheless it seems that there have been far more successful electronic performances of conventional music recently than there has been original music. Part of the reason for this difference is actually one of the limitations of computer music as it is produced in most situations today: the lack of immediacy between performance time (at which the input is fed to the computer) and audition time (when the sound is produced). Very few installations today have any immediate playback facilities except for more than a few simple sounds, and real-time computer music synthesis of entire compositions is practically impossible (see reference 3, chapter 9).

The inherent difficulty in producing sounds by computer, together with the delay in hearing the results, forces computer music composers to evaluate their work reflectively and to think carefully before trying things out. While immediacy is important to the act of performance, it is not necessary for the act of composition itself, which is a conceptual task.

An excellent example of the kind of creativity on the sound level made possible in computer music is described by J K Randall in his article "Operations on Waveforms" (Perspectives of New Music, volume 5, number 2, spring/summer 1967, page 124). These ideas are exploited compositionally in Randall's Lyric Variations for Violin and Computer, written in 1968 and recorded on Vanguard C-10057. Randall describes several original ideas he used to synthesize new timbres from completely new theoretical principles. He produces sounds in which the individual constituent partials (harmonics of the fundamental) of a tone are operated upon just as the pitch, rhythm, and dynamics are in conventional music. The result is music of unusual interest in which all variations of the sounds relate in novel ways to the structure of the music.

The same kinds of processes in which partials are treated as pitches were employed in my compositions entitled Studies in Timbre, of which there are presently four. The first study employs sounds that dissolve into others by glissandos that move from one partial to another. These are contrasted with other sounds of fixed pitch but changing timbre. The second study uses instrument-like sounds with transient elements in the attack and variations in the steady state portions of the tones. The third is based on varying timbral patterns associated with specific musical events. The typical instrument produces 12 partials that fade in and out in different ways over the course of each tone, correlated to amplitude and location changes. Contrasts between partials of some tones and the fundamental frequencies of others are emphasized in conjunction with rhythmic, dynamic, and harmonic properties that develop concomitantly. The fourth study uses harmonic series that progress to non-harmonic ones, but which nevertheless preserve some abstract ratio between the elements, thus producing another kind of dissolution of a tone into a somewhat clangorous sound.

The main point of these considerations is to show how a computer music composer is encouraged to experiment with original ideas that often lead to results unobtainable by any other method of music production. But since computers are theoretically capable of producing any sounds, this is not the only beneficial or distinguishing result. The fact is that the same sounds can be produced by different descriptions that are based on generalizations about different aspects of the sounds. Thus, the really important point is that it is the relationship between the input and the output which is clarified by the process of computer music synthesis. Whatever this may be in relation to a specific musical work is in itself a concept about the structure of that work.

Musical Composition

Although I have emphasized the creativity involved in the other disciplines discussed, it is obvious that composition is, by comparison, the ultimate creative act to be delegated to a computer. Many people express disbelief or doubt that this can really be done successfully. I believe that this doubt is rooted more in misunderstanding of what musical composition itself involves rather than of what the computer does. Composers may employ mysterious methods, but they are even more prone to making mysterious explanations of straightforward methods in order to preserve their compositional mystique.

Any detailed consideration of musical analysis or theory suggests numerous avenues of approach to the compositional method employed. Whether a piece actually has been composed according to the principles that can be abstracted from its structure is not necessarily relevant, and often can never be known. In recent years, the work of Heinrich Schenker and post-Schenkerian analysts has attempted to discover large scale structural properties in tonal music which almost certainly were not consciously considered by the composers who wrote the music, and which raise questions not previously posed in the history of theory.
UCSD PASCAL, the powerful, general purpose language developed for large and complex programs is now available for your TRS-80. This new, Industry standard language avoids many errors normally encountered when using older languages.

Although it is relatively easy for beginners to learn and use, it provides added potential to cover most problems of practical programming. The UCSD PASCAL Compiler allows long variable names, and block structured control statements.

The FMG/UCSD PASCAL System opens a new generation of value for your TRS-80.

<table>
<thead>
<tr>
<th>Package Includes</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>$150.00</td>
</tr>
<tr>
<td>Screen Editor</td>
<td></td>
</tr>
<tr>
<td>Z80 Macro Assembler</td>
<td></td>
</tr>
<tr>
<td>Debugger</td>
<td></td>
</tr>
<tr>
<td>Pascal Compiler</td>
<td></td>
</tr>
<tr>
<td>Utilities and System</td>
<td></td>
</tr>
<tr>
<td>Reference Book</td>
<td></td>
</tr>
</tbody>
</table>

Available without Macro Assembler Linker and Debugger $100.00

CBASIC
A commercially oriented BASIC compiler-interpreter designed for use with the CP/M disk operating system.
The compiler features:
- Variable Length Files
- Double Precision Numbers
- Optional Line Numbers
- "While...End"
- Structural Group
- Long Variable Names (up to 31 characters)
- Multiline User Defined Functions
- 98 Page Manual

$99.00
(Manual only $25.00)

FORTRAN
Comparable to compilers on large mainframes and minicomputers. All of ANSI Standard FORTRAN X3.9-1966 is included except COMPLEX data type. Therefore, users may take advantage of all the many applications programs already written in FORTRAN.

Package includes:
- FORTRAN Compiler
- Macro Assembler (Z80)
- Linker
- Library
- Lib Manager (Not in TRS-DOS version)

Price $350.00
Manual $25.00
(Specify TRS-DOS or CP/M versions)

MACRO ASSEMBLER
A relocatable Macro Assembler for both 8080 and Z80 op-codes. Incorporates most big computer assembler features without sacrificing speed or memory size. Package includes:
- Comment Blocks
- Full Intel Standard Macros
- Lib Manager (not included in TRS-DOS version)
- Linker
- Library

$219.00 (Included in FORTRAN)
Manual $25.00
(Specify TRS-DOS or CP/M versions)

TEX\text{\textsc{EXT}} FORMATTER
Powerful text formatting capability using CP/M Editor. TEX

Diskette and Manual $125.00
Manual $25.00

肋

SID\text{\textsc{s\textit{I}}} SYMBOLIC INSTRUCTION DEBUGGER
- Symbolic Memory
- Reference
- Built-In Assembler/Disassembler
- SID Diskette and Manual

$125.00
Manual $25.00

Call or Write for Complete Information

FMG for HIGH LEVEL LANGUAGES
BASIC FORTRAN COBOL and now PASCAL
CP/M is a registered trademark of Digital Research Corp.
TRS-80 is a registered trademark of Radio Shack.

Circle 134 on inquiry card.
Computer composition can be employed as a method of verifying theories and analyses of specific works, especially if they consist of multilayered abstractions like Schenkerian analysis. These abstractions suggest a generative approach starting from a background and building through successive layers of elaboration to the foreground. One method of verifying such an analysis is to prove that it can actually regenerate the work, at least in its essential structural aspects, by a particular sequence of operations. Another method is to change some of the background or middleground structure or the particular sequence of operations to produce new works that can be compared with the original. Although they might not be intended to be taken seriously as original compositions (since their structure is derivative of another work), such compositions would unquestionably be computer composed music.

Instead of using this structural approach, early work in computer composition has tended to emphasize the sensational aspects of “machine music” and has been based on random and aleatoric methods (see reference 2). Very few conditions of the intended music have been specified, with the result that certain characteristics are quite clearly present (eg: consonances) or absent (eg: dissonances) but the music is otherwise aimless and without structure. These facts are quite obvious when the music produced resembles some early style like a Bach invention, but are less clear when some avant-garde style is imitated. Nevertheless, such experiments (often designated as experiments rather than as music) are not typical of musical compositional methods in general, nor of computer work in this field.

In my own music I have employed computer composition in extensive ways (“Composing by Computer,” Computers and the Humanities, volume 9, 1975, page 281).

Nevertheless, the computer does not make any decisions that I would not make myself, nor indeed that I have not already made when it executes my instructions; it merely carries out many time-consuming calculations that would otherwise have to be done by hand. Basically, the program works out aspects of the foreground syntax which assure various rhythmic, harmonic and structural properties. All aspects of this syntax are specified by instructions provided together with the data (pitch classes) on which they are carried out.

Using a program like this allows composers to work more from a background perspective than from the foreground. They are able to concentrate on the large scale structural properties without being encumbered with the foreground details that may arise from these characteristics; these are handled quickly and automatically by the computer. Several different possibilities for working out a given passage can be tried before any commitment is made to them.

Even more important, the process of writing a program to carry out such compositional details forces composers to be absolutely explicit about their procedures and intentions. Any errors or incompleteness in the specifications will be exposed when a program is executed by the computer. Ad hoc methods that composers may use to fix certain passages when they don’t work out as expected are not necessary, because it is easy enough to revise the program once these problems are exposed. The composer’s attention is thus always directed to the most important conceptual aspects of the music, and his or her ability to solve problems in these areas is facilitated.

Conclusion

By examining aspects of the four musical disciplines discussed above, we have seen many instances where the use of computers...
MicroSource™ is becoming the chosen business software producer. For all the right reasons. For example, before you ever see one of our applications, it's field tested and refined. Another reason is our Operations Manuals. They're called the finest in the industry ... you never need special training to use MicroSource. One more reason: versatility. MicroSource works with different terminals and printers. You can use your favorite terminal, and choose between speed or print quality in a printer. Add in the rest of the story — like support, enhancements, low cost — and you get the picture.

If your dealer stakes his business on MicroSource software, can you afford to do less?

MicroSource™

MoneyBelt — The Money Manager™
A modular in-house bookkeeping package with general ledger, accounts payable and receivable, and payroll. Buy at your own pace ... one at a time or all at once. Less than $8000 (under $200/mo. on a financing lease) for all hardware and first module (sugg. list). Horizon and other North Star disk based systems.

LedgerPlus — The Company Bookkeeper™
Also modular, with all the accounting modules of MoneyBelt, plus inventory, check reconciliation and mailing list. Each interacts with the others, but stands alone, too. A first for Vector Graphic, MZ, Apple and Micropolis disk based systems. Less than $8000 (under $200/mo. on a financing lease) for all hardware and first module (sugg. list).

Other business software packages by MicroSource: AutoScribe — The Paperwork Manager™, Bookkeeper — The Office Accountant™, TimeKeeper — The Time Accountant™.

MicroSource
1425 W. 12th Place • Tempe, Arizona 85281 • (602) 894-9247 • Cable: MICROAGE
Telex: 165 033 (MICRO AGE TMPE)
The following are trademarks of The Phoenix Group, Inc.: MicroSource, AutoScribe, Bookkeeper, MoneyBelt, LedgerPlus, TimeKeeper.
DEALERSHIPS STILL AVAILABLE

Circle 227 on inquiry card.
necessitates a conceptual approach to the questions at issue. While many projects continue to pose problems because of the difficulty of computer programming, and of preparation of data, progress in these fields is nevertheless being made in significant ways. There still remain, and there always will be, gaps between those subjects that computer researchers tend to deal with and those that traditional, noncomputer people will choose, partly in the belief (from both sides) that some subjects are not suitable or possible for computer analysis. While important questions are always difficult to answer and sometimes require generations of work in order to achieve accurate answers, they are never impossible. We will know that we have truly reached a high state of conceptual thinking when the conclusions of computer research tend to agree with those of noncomputer research—both will then be operating on the same level.

GLOSSARY

Aleatoric music: music, usually composed by computer, in which all or many of the most important characteristics are chosen randomly.

Analysis: description of music according to certain fundamental properties that are judged to be relevant to a given piece, showing how different elements in the work may be related by these properties.

Atonality: music which is not tonal, and where a specific attempt is made to avoid reference to a key, or when the concept of key is not relevant. The term has two basic uses, specific and general: specifically, it describes a body of early twentieth century music by Schoenberg, Berg, Webern, and others, preceding 12 tone music (qv), in which tonality (qv) was consciously avoided. Generally, the term is used to describe any twentieth century music avoiding tonality that cannot be described simply by other methods.

Background: musical substructure describing large scale properties and relations that may not be evident from an inspection of the immediate note to note properties, or from a superficial auditioning of the music; see foreground. In certain theories, the background is considered divisible into several levels.

Computer music: music employing computers at any stage of its composition or realization as sound.

Cycle of fifths equivalence: an operation on a group of notes in which each element is replaced by its equivalent on the circle of fifths; analogous to inversion (qv). The circle of fifths generates the total chromatic (qv) by starting on any note, adding the note a perfect fifth (seven half steps) higher, and continuing this process.

Electronic music: music in which the sounds are produced by electronic music synthesizers (qv); and (4) computer music (qv), in which the sounds are produced or controlled by computers.

Encoding language: a method in which musical notation may be represented in code suitable for computer input. The most widely used and documented languages to date are IML (Intermediary Musical Language), DARMS (Descriptive Alphanumeric Representation of Musical Symbols), the "Plaine and Easie Code System for Musicke," LMT (Linear Music Transcription), and ALMA (Alphanumeric Language for Music Analysis).

Foreground: the "surface" of a piece of music, including sounds that are simultaneous or that appear in direct succession; distinguished from the background (qv).

Glissando: a continuous sliding from one pitch to another.

Half step: the smallest interval (qv) used in music based on equal temperament (the tuning system in widespread use in Western cultures since the eighteenth century). Music employing smaller intervals is said to be microtonal.

Harmony: a theory describing properties of simultaneous sounds (chords) in tonal music (see tonality). Chords are expected to move in certain progressions, and dissonances resolve into consonances according to various rules.

Interval: the distance, measured in half steps, between two pitches or pitch classes. Tonal music also employs another definition of interval, based not on the sound but on the notation for the two notes involved.

Inversion: an operation on a group of notes in which each element is replaced by its equivalent on the descending chromatic scale, or ascending circle of major sevenths (11 half steps). (In this formulation, an ascending chromatic scale would be the identity operation.) Identity and inversion, along with cycle of fifths equivalence and its inversion (cycle of fourths equivalence) are the only single interval cycles that generate the total chromatic (qv).

Octave: a musical interval of 12 half steps, corresponding to the frequency ratio of 2 to 1. Pitch-related octaves possess a strong similarity, which has been called octave equivalence.
Pitch: a single tone in a musical composition. Most pitches used in music are taken from the range of the 88 produced by the piano, but some extend beyond this range.

Pitch class: a group of pitches separated by any number of octaves. In musical theories, pitches in different octaves employ the same letter names (C, C♯ or Db, D, D♯ or Eb, etc) reflecting the fact that theories are based on pitch classes rather than pitches. Pitch classes are also often called notes. While there are many pitches, there are only 12 distinct pitch classes in Western music.

Pitch structure: a set of pitch class collections that all possess the same intervallic structure, so that they are related by transposition (qv). Pitch structure is the basic way that collections of tones, such as chords, are compared: as major triads, minor triads, etc.

Spectrum: the overtone structure of a sound, represented as a series showing the amplitude of each overtone present; see timbre.

Structure: any abstract method in which the properties of a piece of music can be encompassed. Generally, a piece is divided into several sections, each of which has a different structure. Sometimes structure is described in terms of a function or purpose at work in an entire section, such as introduction, development, or statement (of a theme or idea). Sometimes structure is described numerically, proportionally, or in other abstract ways.

Synthesizer: a machine that generates and processes sounds automatically, used in the production of electronic music. Most synthesizers consist of a number of discrete components that perform different functions (eg: oscillators that generate tones, filters that modify their spectrum, etc). Some recently designed synthesizers include microprocessors, which are used as controlling devices.

Theory: a set of concepts describing properties and relations that can be shown to exist in a body of musical literature. The most commonly known theories today include tonality (qv) and 12 tone music (qv), but there are many others that have been described and used by various authors. Most theories have originated after the music they purport to describe has been written. Speculative theory originates as speculation, before such music has been written, so that its success or failure has not yet been demonstrated.

Timbre: musical tone quality, descriptive of the way in which different tones may possess a similarity not on the basis of pitch, amplitude, or rhythm. In electronic music, this term is normally used synonymously with spectrum (qv), but in contexts where it is described qualitatively rather than numerically.

Tonality: a theory describing music which is in a key, or a series of keys, usually pertaining to music written during the eighteenth and nineteenth centuries. Basic concepts include the major and minor scales, triads, and specific rules according to which dissonances resolve into consonances. Much of the music of the twentieth century is based on an extended notion of tonality, in which some, but not all, of the basic concepts are employed.

Total chromatic: any series of notes including all 12 pitch classes.

Transposition: the addition or subtraction of a constant interval to each tone in a collection, moving the set up or down by a uniform amount.

Triad: a chord consisting of three notes (or pitch classes) with a root note, and other notes a third and fifth above the root.

12 tone music: music in which all note successions, and sometimes other properties are based on sets of orderings of the total chromatic.

REFERENCES
Nybbles

Dr G Louis
OB/GYN Department
St Michael’s Hospital
30 Bond St
Toronto CANADA M5B 1W8

“Tiny” Pascal in 8080 Assembly Language

The p-code interpreter, Pascal to p-code compiler, and p-code to 8080-code translator described in “A Tiny Pascal Compiler” by Kin-Man Chung and Herbert Yuen (September thru November 1978 BYTE) have been rewritten in 8080 assembly language. In addition to providing approximately two orders of magnitude increase in speed, the object versions run in far less memory.

To use the assembly language version of tiny Pascal you need an 8080 (or Z-80) microcomputer system with not less than 12 K bytes of memory (the package would run in 8 K bytes but you need file space), a copy of the articles by Chung and Yuen, which have been reprinted by BYTE magazine and are available for $3 (BYTE document number 100), an elementary acquaintance with 8080 assembly language; the package of assembly language modules.

Additionally, it is desirable to have a text editor, an assembler that stores lines in the form: length byte, 4 digit line number, space or tab, text, carriage return, and a relocator program that is useful for tailoring the package to your system.

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 “Tiny Pascal in 8080 Assembly Language” (document #111) has been added to the Nybbles Library. Use the coupon below to order your personal copy, at $10.00 (US and Canada) or $12.00 (foreign air-mail) postpaid.

Please send __ copies of BYTE Nybble #________ at $______ postpaid.

Please remit in US funds only.

____ Check Enclosed

Bill my VISA # __________________________ Exp Date __________

Bill my MC # __________________________ Exp Date __________

Name __________________________

Street __________________________

City __________________________ State __________ Zip Code __________

BYTE Nybbles Library, 70 Main St, Peterborough NH 03458

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

---

**IMMEDIATE DELIVERY**

Domestic & Export

**DEC LSI -11 COMPONENTS**

A full and complete line with software support available.

Mini Computer Suppliers, Inc.

25 CHATHAM ROAD

SUMMIT, NEW JERSEY 07901

SINCE 1973

(201) 277-6150 Telex 13-6476

---

**the ULTIMATE in CHEAP VIDEO**

BOOK & KIT

ONLY $42.95

Don Lancaster’s “Cheap Video” concept allows almost unlimited options, including:

- Scrolling: Full performance cursor.
- Line/Character formats of 16/32, 24/80, 32/64 .... or almost anything.
- Graphics – up to 256 X 256 B&W; 96 X 128 COLOR (requires low-cost option modules)

Works with 6502, 6800 and other micros.

SPECIAL OFFER: Buy the Kit (uppercase alphanumeric option included) & get the Book at 1/2 price.

I’m Sold. PLEASE RUSH......

( ) SEND FREE CATALOG

( ) TVT-658 Kit & Cheap Video Cookbook $42.95 (enclosed)

( ) TVT-658 Kit only (book required for assembly) $39.95

name____________________

address:___________________ state:__________ zip:_______

**MIA ELECTRONICS, DEPT 7-B, 1030 W. WILSHIRE BLVD, OKLAHOMA CITY, OK 73118**

Circle 236 on inquiry card.

---

July 1979

BYTE Publications Inc 175

Circle 297 on inquiry card.
You are at one end of a vast hall stretching forward out of sight to the west. There are openings to either side. Nearby, a wide stone staircase leads downward. The hall is filled with wisps of white mist swaying to and fro almost as if alive. A cold wind blows up the staircase. There is a passage at the top of the dome behind you. Rough stone steps lead up the dome.

> GO DOWN

You are in the hall of the Mountain King, with passages off in all directions. A huge green fierce snake bars the way!

> ATTACK SNAKE

Attacking the snake both doesn't work and is very dangerous.

> FEED SNAKE

There's nothing here it wants to eat (except perhaps you).

> YELL

Are you trying to somehow deal with the snake?

> YES

I am prepared to give you a hint, but it will cost you 2 points. Do you want the hint?

The preceding paragraphs are taken from a session I played on a computer running a program called Adventure, which has used many, many cycles of machine time on many, many computers in the past year or two. The original version, I believe, was written in FORTRAN, by Willie Crowther (now at Xerox Palo Alto Research Center), and was subsequently modified and rewritten by Don Woods at Stanford's Artificial Intelligence Laboratory.

The version I ran has been bootlegged through several generations, and I'm not sure of its exact ancestry. It requires close to 200,000 bytes of memory on an IBM 370/168. Adventure has been translated into PL/I, APL, and BASIC. It is the successor to Hunt the Wumpus and the many Star Trek games.

I hope it is the precursor of more elaborate games which combine computers with fantasy to produce an "electric novel," which the user and the computer write or experience together. Already, a few computers around the country are offering a child of Adventure called Zork or Dungeon, which has a more sophisticated understanding of English, and a whole new set of problems to solve and monsters to defeat. Space War, which used to belong to the "freaks" and the "hackers" (in the hours after the managers went home), is now available in your neighborhood tavern for 25¢ per enemy starship. How much longer will it be until we can each rule our own kingdom and rescue our own fair maidens?

There aren't many personal computers yet with 200 K bytes of memory available, and not all of us have free or inexpensive access to the machines on which Adventure can run. A smaller computer would require a floppy disk file for keeping the cave descriptions, and most users have no hardware for
YOU’VE READ THEM ALL—NOW GET THE BEST!
The 80-US Journal
A PUBLICATION FOR TRS-80 USERS - COVERS ALL TRS-80 CAPABILITY!

BI-MONTHLY, SEND $16.00 CHECK OR MONEY ORDER FOR A 1 YEAR SUBSCRIPTION TO:
80 US
PO Box 7112
TACOMA, WA. 98407

MASTER CHARGE, VISA OK - (206) 759-9642
WANT TO CHECK IT OUT FIRST? SEND $3.00 for current sample copy.

RESEARCH®
P.O. Box 579 • Pacific Grove, California 93950
(408) 649-3896

NEW SHAPES
by Julius Guest
Is a collection of 110 original and fascinating computer-generated designs created by the author over an eight-year period. The designs are satisfying art creations in themselves and may evoke the responses that their titles suggest. As each art piece is accompanied by a mathematical formulation and program, the reader may thus create his own ‘New Shapes’ and derive as much enjoyment from them as the author obviously did.

The designs are reproduced in color (some in exciting solid reverse) on high quality art paper. Details: 174 pages, 28cm x 27cm (11" x 10 1/2").

(Total $12.00 U.S., that's only $1.16 per diagram!) Add $1.25 postage. ISBN 0-9596384-INTRODUCTION TO PL/I & STRUCTURED PROGRAMMING

This book covers an easy-to-learn subset of ANSI PL/I. Fundamentals are emphasized and structured programming is encouraged. The book contains numerical and non-numerical worked examples, and 200 exercises and problems. The versatile subset is defined in only 3 pages of syntax diagrams. Suted to a one- or two-semester course. Also appropriate for self-study. An excellent reference document—only $5.00 U.S. (Add 80c postage.)

Contents: ten chapters and six appendixes. Data representation, fixed and floating-point, character strings, LIST and EDIT-directed I/O, WHILE, DO, probes, procedures and functions, built-ins, computational error detection and correction, ON-units.

Order direct from publisher R.A. Vowels, 93 Park Drive, Parkville 3052, Australia. Personal checks accepted in your currency, or Bank Draft Check in Aust. currency. Discount: 5% for 2 or more books.

Circle 104 on inquiry card.
Circle 142 on inquiry card.
Circle 95 on inquiry card.
Circle 383 on inquiry card.
Circle 383 on inquiry card.

CP/M® LOW-COST MICROCOMPUTER SOFTWARE

CP/M® OPERATING SYSTEM:
• Includes Editor, Assembler, Debugger and Utilities.
• Standard version for 8080, Z80, or Intel MOS (other versions available.)
• For IBM-compatible floppy discs.
• $100-Diskette and Documentation.
• $25-Documentation (Set of 6 manuals) only.

MAC™ MACRO ASSEMBLER:
• Compatible with new Intel macro standard.
• Complete guide to macro applications.
• $90-Diskette and Manual.

SID™ SYMBOLIC DEBUGGER:
• Symbolic memory reference.
• Built-in assembler/disassembler.
• $75-Diskette and Manual.

TEX™ TEXT FORMATTER:
• Powerful text formatting capabilities.
• Text prepared using CP/M Editor.
• $75-Diskette and Manual.

DESPool™:
• Background print utility.
• Use with CP/M (version 1.4)
• $50. Diskette and Manual.

DIGITAL RESEARCH®
P.O. Box 579 • Pacific Grove, California 93950
(408) 649-3896

T.D.Q. TAPE DATA QUERY
PET-8K SOL-IIA TRS-80-LEVEL II

FILE MANAGEMENT SYSTEM
—UTILIZES DUAL AUDIO CASSETTE RECORDERS

INTERACTIVE QUERY LANGUAGE
—ENGLISH-LIKE COMMANDS

POWERFUL INFO RETRIEVAL CAPABILITY

COMPUTERIZED BUSINESS & PERSONAL RECORDS
—CUSTOMIZE YOUR OWN FILE STRUCTURES
—CREATE & MAINTAIN DATA FILES
—NO PROGRAMMING EXPERIENCE REQUIRED

IMPLEMENTED IN BASIC
T.D.Q. CASSETTE WITH MANUAL & REF. CARD $50.00
THE FOLLOWING PRE-DEFINED T.D.Q. FILE STRUCTURES ARE AVAILABLE TO SOLVE YOUR DATA PROCESSING NEEDS:

INVENTORY CONTROL $35.00
ACCOUNTS RECEIVABLE $35.00
ACCOUNTS PAYABLE $35.00
ORDER PROCESSING $35.00
CUSTOMER DIRECTORY $25.00
APPOINTMENT SCHEDULING $25.00

EACH WITH CASSETTE AND MANUAL
SEND SELF-ADDRESSED STAMPED ENVELOPE FOR COMPLETE SOFTWARE CATALOGUE.
SEND CHECK OR MONEY ORDER TO:
H. GELLER COMPUTER SYSTEMS
DEPT. B, P.O. BOX 350
NEW YORK, NY 10040

NEW YORK RESIDENTS ADD APPLICABLE SALES TAX!
 ridden fast random access outside the computer memory. Adventure on everybody’s computer is still in the future, although it is coming soon.

I was playing Adventure at about the time Peninsula School, Menlo Park CA, received two Commodore PET 2001 machines for the children to work with; and the incredible idea of Adventure on a PET was too exciting to ignore. Adventure on a PET, with only 7123 bytes available for the BASIC program, is impossible, but couldn’t I do something just a bit less ambitious? In a couple of intense work weeks, I wrote a program I called Quest, which ran on the PET.

Comparing Quest and Adventure

Compared to Adventure, Quest is a toy, in the same measure that the PET is a toy compared to larger computer systems. But it is an enjoyable and even exciting toy, in the same ways that the PET can be enjoyed by someone who can also play with the big computers. Each system has its own problems and pleasures. Adventure, as you can see from the opening paragraphs, has some novel problems for the adventurer to solve, and will proceed according to the adventurer’s 2 word commands. On the other hand, the problem you must solve in Quest is basically to find your way around the cave. The only commands Quest normally understands are six directions: NORTH, SOUTH, EAST, WEST, UP, and DOWN, and even there, only the first letter is examined.

No huge green snake will confront you, and even the pirate, who swoops down to protect his treasure at some point, is beyond your control. He steals back the treasure after you have found it, and the problem continues: find the treasure again, and find your way out of the cave. To make it more interesting, various passages open and close according to your progress through the game.

One limitation which Quest and Adventure share is that they never change. Once you know how to get past the snake, you always know, and once you can find the treasure, you always know where it is. A friend of mine has suggested having earthquakes, which open and close passages at random. It seems to me, however, that that problem continues: find the treasure again, and find your way out of the cave. To make it more interesting, various passages open and close according to your progress through the game.

One limitation which Quest and Adventure share is that they never change. Once you know how to get past the snake, you always know, and once you can find the treasure, you always know where it is. A friend of mine has suggested having earthquakes, which open and close passages at random. It seems to me, however, that that simply makes a bigger problem of the same kind, and I would rather have different problems. In that respect, both Adventure and Quest are very limited.

In a closer approach to the electric novel, there would be no guarantee that the problems can really be solved. In a Star Trek game, for instance, the fate of the Enterprise depends on the random number generator,
which can set the starship down in the same quadrant with four Klingon commanders and two super commanders, or cause all the starbases to be overrun by a plague of tribbles. In Adventure and Quest, as in crossword puzzles, the solution is part of the problem definition, and you know it exists.

It is possible in Adventure to do something which will ruin your chance for success. For instance, if you feed the bird to the snake, you will never get past the problem to which the bird is the solution. However, this doesn't change the basic limitation.

An important part of both games is the descriptions of the locations. These are of course not created by the computer, but were elaborated over a long period by the programmer and his friends. The topology of the Quest cave could be more complex if the location descriptions were something like "YOU'RE AT LOCATION 28. NOW WHAT?", but that would spoil a good part of the game. The descriptions in Quest have been worked out very carefully. Some of them are just for fun, and some of them have hints about the neighborhood.

Scoring

The original version of Quest had no scoring at all, to minimize the competitive situ-

```
Listing 2: Game of Quest in BASIC.
1 REM QUEST BY ROGER CHAFFEE
2 REM INSPIRED BY WILL CROWTHER'S "ADVENTURE"
3 REM COPYRIGHT (C) 1978 PENINSULA SCHOOL, MENLO PARK, CA
4 REM PERMISSION TO USE, NOT TO SELL
5 REM THE ORIGINAL VERSION OF THIS PROGRAM HAS BEEN WRITTEN ON A
6 REM COMMODORE PET 2001. THIS VERSION HAS BEEN CONSIDERABLY
7 REM RECODED, AND IS IN "PLAIN VANILLA" BASIC, WITH THE
8 REM EXCEPTION OF THE RANDOM NUMBER GENERATOR IN LINES
9 REM 6600-6800, THE STRING MANIPULATION, AND THE IF ... THEN
10 REM STATEMENTS WHICH GIVE A STATEMENT TO PERFORM INSTEAD OF
11 REM A STATEMENT NUMBER TO GO TO.
12 REM VARIABLES USED
13 REM N NODE (CAVE) NUMBER
14 REM M MOVE COUNTER
15 REM T CURRENT LOCATION OF TREASURE (-1 FOR CARRYING)
16 REM U1, U2 FIRST AND SECOND HIDING PLACES (NODE NUMBERS)
17 REM W SAVES THE MOVE NUMBER WHEN HE SAID NO, HE DIDNT
18 REM X WANT TO TAKE THE TREASURE WITH HIM.
19 REM Z MAP OF INTERCONNECTIONS.
20 REM N(I, J) IS NEXT NODE FROM NODE J, WHEN YOU GO
21 REM N(I, U1, W, 5 FOR I=1,2,3,4,5,6
22 REM Bounces TO GO TO NODE -2 MEANS BOUNCE BACK TO THE NODE YOU
23 REM CARE FROM.
24 REM FORCED AND/OR RANDOM MOVES
25 REM N(1, W) = 2 MEANS A FORCED MOVE AS SOON AS YOU REACH
26 REM NODE W. IN THAT CASE, N(2, W) OF THE TIME YOU GO
27 REM TO NODE N(3, W). IF YOU DON'T GO THERE, N(4, W)
28 REM OF THE TIME YOU GO TO NODE (5, W), AND THE REST OF
29 REM THE TIME YOU GO TO NODE (6, W).
```

Text continued on page 186
Listing 2 continued from page 179:

35 REM NODE N+100 MEANS NODE N IF YOU DON'T HAVE THE
36 REM TREASURE, AND NODE N+1 IF YOU DO.
37 REM NODE N+200 MEANS NODE N+1 IF YOU HAVE THE TREASURE
38 REM THE SECOND TIME, AND NODE N OTHERWISE.
39 REM NODE N+300 MEANS RUN THROUGH A DELAY LOOP AND THEN
40 REM EXIT CHUTE, AND YOU MAY WANT TO ADJUST THE DELAY
41 REM TIME (LINE 6250).
42 REM
43 REM Q$ INPUT STRING
44 REM A$ CHARACTERS TO MATCH IN THE INPUT ROUTINE
45 REM A2 NUMBER OF CHARACTERS IN A$
46 REM A1 OUTPUT FROM THE INPUT ROUTINE
47 REM P PIRATE FLAG 1 IF PIRATE HAS GOT YOU, 0 OTHERWISE
48 REM N9 SAVES OLD NODE IN MOVE ROUTINE, FOR BOUNCE
49 REM N8 SAVES NODE WE BOUNCED FROM IN MOVE ROUTINE,
50 REM FOR PRINT FLAG
51 REM N0 SAVES OLD NODE IN MOVE ROUTINE, FOR DEAD END
52 REM A0 SAVES OLD DIRECTION IN MOVE ROUTINE
53 REM D DEBUG FLAG (NON-ZERO TO PRINT)
54 REM I,J MISC. COUNTERS
55 REM W TRAVEL FLAG, USED IN SCORING, W(I)=1 IF HE'S
56 REM BEEN TO NODE I, 0 OTHERWISE
57 REM S SCORE
58 REM M9 MAXIMUM NUMBER OF NODES
59 REM D=0
60 REM -------------------------------
61 REM GIVE 'EM SOMETHING TO READ WHILE I GET THE DATABASE SET UP
62 PRINT "QUEST"
63 PRINT "YOU WERE WALKING THROUGH THE"
64 PRINT "WOODS, AND YOU CAME ACROSS THE ENTRANCE"
65 PRINT "OF A CAVE, COVERED WITH BRUSH."
66 PRINT "PEOPLE SAY THAT MANY YEARS AGO A"
67 PRINT "PIRATE HID HIS TREASURE IN THESE"
68 PRINT "WOODS, BUT NO ONE HAS EVER FOUND IT."
69 PRINT "IT MAY STILL BE HERE, FOR ALL I KNOW."
70 READ M,II,T2
71 DIM W(42),M(6,42)
72 REM READ MAP INTO M ARRAY
73 FOR I=1 TO M
74 IF I=N THEN 570
75 PRINT "DATABASE PROBLEM",I,N
76 STOP
77 FOR J=1 TO 6
78 READ M(J,I)
79 NEXT J
80 NEXT I
81 PRINT
82 REM "WHEN YOU ANSWER A QUESTION, I LOOK AT"
83 REM "ONLY THE FIRST LETTER, ALTHOUGH YOU"
84 REM "TYPE THE WHOLE WORD IF YOU WANT."
85 GOSUB 7500
86 REM -------------------------------
87 REM READ MAIN LOOP STARTS HERE
88 REM COUNT MOVES
89 REM MOVE
90 REM CHECK FOR FINDING THE TREASURE
91 REM MOVE
92 REM MOVE
93 REM MOVE
94 REM MOVE
95 REM MOVE
96 REM MOVE
97 REM MOVE
98 REM MOVE
99 REM MOVE
100 REM MOVE
101 REM MOVE
102 REM MOVE
103 REM MOVE
104 REM MOVE
105 REM MOVE
106 REM MOVE
107 REM MOVE
108 REM MOVE
109 REM MOVE
110 REM MOVE
111 REM MOVE
112 REM MOVE
113 REM MOVE
114 REM MOVE
115 REM MOVE
116 REM MOVE
117 REM MOVE
118 REM MOVE
119 REM MOVE
120 REM MOVE
121 REM MOVE
122 REM MOVE
123 REM MOVE
124 REM MOVE
125 REM MOVE
126 REM MOVE
127 REM MOVE
128 REM MOVE
129 REM MOVE
130 REM MOVE
131 REM MOVE
132 REM MOVE
133 REM MOVE
134 REM MOVE
135 REM MOVE
136 REM MOVE
137 REM MOVE
138 REM MOVE
139 REM MOVE
140 REM MOVE
141 REM MOVE
142 REM MOVE
143 REM MOVE
144 REM MOVE
145 REM MOVE
146 REM MOVE
147 REM MOVE
148 REM MOVE
149 REM MOVE
150 REM MOVE
151 REM MOVE
152 REM MOVE
153 REM MOVE
154 REM MOVE
155 REM MOVE
156 REM MOVE
157 REM MOVE
158 REM MOVE
1470 REM TRY THE PIRATE
1480 GOSUB 4000
1490 REM LOOP UNLESS FINISHED
1500 IF T>0 THEN 1400
1510 IF W>5 THEN 1400
1700 REM CALCULATE SCORE
1710 GOSUB 3000
1720 PRINT
1730 PRINT "CONGRATULATIONS! YOU GOT THE TREASURE"
1740 PRINT "OUT IN";ll0;
1750 PRINT "Loves AND YOU GOT"S+10"points!
1760 PRINT "WANT TO HUNT AGAIN? ";
1770 AS="YN"
1771 A2=2
1780 GOSUB 5000
1790 ON A1 GO TO :000.9999.1760
2000 REM ---------------------------------------------
2010 REM FOUND?
2100 IF T<>8 THEN RETURN
2110 IF T<0 THEN RETURN
2120 IF H6+5>50 THEN RETURN
2200 PRINT "DO YOU WANT TO TAKE IT WITH YOU? ";
2210 AS="YN"
2220 A2=2
2230 GOSUB 5000
2240 ON A1 GO TO 2300,2400
2250 PRINT "WELL....."
2260 GO TO 2210
2300 T=1
2310 PRINT
2320 PRINT "OK! LET'S GET OUT OF HERE!"
2330 RETURN
2400 PRINT
2410 PRINT "WE'LL LEAVE IT HERE AND YOU CAN EXPLORE"
2420 PRINT "SOME MORE."
2430 H6=50
2440 RETURN
3000 REM ---------------------------------------------
3010 REM SCORE
3020 S=0
3030 IF T=-1 THEN S=S+5
3040 IF P=1 THEN S=S+10
3050 FOR J=2 TO 119
3060 S=S+II (J)
3070 NEXT J
3080 RETURN
3100 REM ---------------------------------------------
3110 REM PIRATE
3120 IF N=T2 THEN RETURN
3130 IF P=1 THEN RETURN
3140 IF T1<T2 THEN RETURN
3150 IF T<>1 THEN RETURN
3160 REM HE'S AT THE EXIT WITH THE TREASURE. ZAP HIM.
3165 REM (ARGH, HOW DID HE GET HERE, ANYWAY?)
3170 IF N=16 THEN P=160
3180 REM COULDN'T MOVE SINCE HITTING TIGHT TUNNEL WITH TREASURE
3190 IF P=0 THEN P=P+1
3200 IF N=3 THEN P=P+1
3210 REM GIVE HIM A FEW MORE MOVES, THEN ZAP HIM
3220 IF P<15 THEN RETURN
3230 PRINT
3240 PRINT "SUDDENLY THE PIRATE LEAPS OUT OF THE"
3250 PRINT "GLOOM AND GRABS THE TREASURE FROM YOU!"
3260 PRINT "HAM!", HE SHOUTS, "YOU FOUND ME!
3270 PRINT "TREASURE, DID YOU?! WELL, I'LL HIDE"
3280 PRINT "IT BETTER THIS TIME!!"
3290 PRINT "AND HE DISAPPEARS INTO THE DARKNESS"
3300 PRINT "WITH THE TREASURE."
3310 p=1
3320 T=T2
3330 RETURN
4000 REM ---------------------------------------------
4010 REM INPUT
4020 IF N=T2 THEN RETURN
4030 IF P=1 THEN RETURN
4040 IF T1<T2 THEN RETURN
4050 IF T<>1 THEN RETURN
4060 REM HE'S AT THE EXIT WITH THE TREASURE. ZAP HIM.
4065 REM (ARGH, HOW DID HE GET HERE, ANYWAY?)
4070 IF N=16 THEN P=160
4080 REM COULDN'T MOVE SINCE HITTING TIGHT TUNNEL WITH TREASURE
4090 IF P=0 THEN P=P+1
4100 IF N=3 THEN P=P+1
4110 REM GIVE HIM A FEW MORE MOVES, THEN ZAP HIM
4120 IF P<15 THEN RETURN
4130 PRINT
4140 PRINT "SUDDENLY THE PIRATE LEAPS OUT OF THE"
4150 PRINT "GLOOM AND GRABS THE TREASURE FROM YOU!"
4160 PRINT "HAM!", HE SHOUTS, "YOU FOUND ME!
4170 PRINT "TREASURE, DID YOU?! WELL, I'LL HIDE"
4180 PRINT "IT BETTER THIS TIME!!"
4190 PRINT "AND HE DISAPPEARS INTO THE DARKNESS"
4200 PRINT "WITH THE TREASURE."
4210 p=1
4220 T=T2
4230 RETURN
5000 REM ---------------------------------------------
5010 REM INPUT
5020 REM FIRST CHARACTER OF AT INPUT STRING IS COMPARED WITH
5030 REM THE LETTERS OF AS, AND IF THERE IS A MATCH, THE INDEX
5040 REM IN AS IS RETURNED IN A1. IF NO MATCH, SIZE(AS)+1 IS
5050 REM RETURNED.
5060 REM GET INPUT STRING
5070 INPUT QS

Listing 2 continued on page 182
Listing 2 continued from page 181:

5080 REM USE ONLY FIRST CHARACTER
5090 QS = LST(QS,1)
5100 REM SEARCH FOR THE CHARACTER QS IN THE STRING AS. IN THIS VERSION
5110 REM OF BASIC, NDX IS THE INDEX FUNCTION, WHICH DOES EXACTLY THAT.
5120 A1=NDX(AS,QS)
5130 REM BUT CHECK FOR THE CASE WHERE THE CHARACTER WAS NOT FOUND
5140 IF A1=0 THEN A1=A2+1
5145 RETURN
5150 REM IF YOUR VERSION OF BASIC DOESN'T HAVE THE NDX FUNCTION, BUT
5160 REM DOES, FOR INSTANCE, HAVE A FUNCTION WHICH WILL PICK A
5170 REM PARTICULAR CHARACTER FROM A STRING, SUCH AS MID(AS,A2,1)
5180 REM PICKING THE A2-TH CHARACTER FROM AS, YOU MIGHT USE THE
5190 REM FOLLOWING CODE.
5200 REM FOR A1=1 TO A2
5210 REM IF QS=MID(AS,A1,1) THEN RETURN
5220 REM NEXT A1
5230 REM A1=A2+1
5240 REM RETURN
6000 REM -----------------------------~-----------------------
6010 REM LOVE
6020 REM REMEMBER WHERE WE ARE, FOR BOUNCE.
6030 N9=N
6040 REM SET N9 TO ANYTHING BUT YOU CAN'T GO THAT WAY
6050 N9=0
6060 REM ASK WHICH WAY
6070 GOSUB 7000
6080 REM REMEMBER WHERE WE ARE, UNLESS A DEAD END
6090 IF N=1 THEN 6120
6100 N=0
6110 A0=A1
6120 PRINT
6130 I=N(A1,N)
6140 IF I=-2 THEN I=N9
6150 IF D<>0 THEN PRINT "DEBUG";N;"TO":I
6220 IF I<500 THEN 6300
6230 REM DELAY LOOP TO WASTE SOME TIME
6240 I=I-500
6250 FOR J=0 TO 100
6260 NEXT J
6270 GO TO 6200
6300 ON I/100 GO TO 6340,6370
6310 REM NORMAL ROUTE--LESS THAN 100
6320 N=1
6330 GO TO 6400
6340 REM N+100. ADD ONE IF CARRYING THE TREASURE
6350 N=I-100
6355 IF T=-1 THEN N=N+1
6360 GO TO 6400
6370 REM N+200. ADD 1 IF CARRYING TREASURE THE SECOND TIME
6380 N=I-200
6390 IF T=-1 THEN N=N+P
6400 IF N=1 THEN 6500
6410 REM DEAD END. TURN IT SO YOU GET OUT THE OTHER WAY
6420 FOR J=1 TO 6
6430 H(J,1)=2
6440 NEXT J
6450 M(7-A0,N)=N9
6500 REM PRINT OUT THE NODE DESCRIPTION
6510 IF N<2 THEN GOSUB 8000
6520 REM REMEMBER WE'VE BEEN HERE
6530 W(N)=1
6540 N9=N
6600 IF M(1,N)<-2 THEN 6800
6610 REM FORCED MOVE, WITH RANDOM DESTINATIONS
6620 REM ON THIS VERSION OF BASIC, J=-1 FOLLOWED BY RND(J)
6630 REM GETS YOU A NUMBER BETWEEN ZERO AND ONE.
6640 REM YOUR VERSION WILL DIFFER, AND THE NEXT FIVE
6650 REM LINES WILL HAVE TO BE CHANGED.
6660 I=N(6,N)
6670 J=1
6680 IF M(4,N) > 100*RND(J) THEN I=N(5,N)
6690 J=1
6700 IF M(2,N) > 100*RND(J) THEN I=N(3,N)
6710 IF D<>0 THEN PRINT "DEBUG BOUNCE TO":I
6720 REM NOW HAVE A NEW DESTINATION. GO BACK AND REDO IT
6730 GO TO 6200
6800 RETURN
7000 REM -----------------------------~-----------------------
Circle 53 on inquiry card.

Announcing...

THE GUIDE TO DESKTOP
MICROCOMPUTER SELECTION

A guide to the selection of a desktop computer that is right for your needs. Over 100 pages are crammed with features & specifications on models offered by the leading suppliers in the industry today. Manufacturers such as Radio Shack, Apple, Wang, Tektronix, IBM & Hewlett-Packard are included.

Other features include:

- a matrix of BASIC statements with over 250 entries comparing the capabilities of 8 different BASIC Interpreters.
- a matrix of hardware characteristics comparing desktop systems and their associated options.
- capabilities to consider when determining your needs & options.
- space within each matrix to include your choice of system not listed.
- key elements for development of software which can convert from one computer to another.

TO OBTAIN YOUR COPY, RUSH $13.65 PLUS $1.35 POSTAGE AND HANDLING TO:

Atlantic Analysis Corporation
5 Koger Executive Center • Suite 219 • Norfolk, Virginia 23502
(804) 461-1980
VA Residents, add 4% sales tax

BYTE July 1979 183
Listing 2 continued from page 182:

7100 PRINT
7110 PRINT "WHICH WAY?"
7120 AS="NEDWSP"
7130 A2=7
7140 GOSUB 5000
7150 IF A1<8 THEN 7300
7160 PRINT "WHICH WAY DO YOU WANT TO GO?"
7170 REM GIVE INSTRUCTIONS
7180 GOSUB 7500
7190 REM DESCRIPT THE LOCATION AGAIN
7200 GOSUB 8000
7210 IF A1<7 THEN 7400
7220 BEG GIVE INSTRUCTIONS
7230 GOSUB 12500
7240 REM START AGAIN
7250 GO TO 7100
7300 IF A1<7 THEN 7400
7310 BEGIN CALCULATE AND PRINT SCORE
7320 GOSUB 3000
7330 PRINT "YOU HAVE"S "POINTS!"
7340 REM START AGAIN
7350 GO TO 7100
7400 RETURN
7500 REM -------------------------
7510 REM SUBROUTINE TO GIVE INSTRUCTIONS
7520 PRINT
7530 PRINT "TYPE N,S,E,W, OR D FOR NORTH, SOUTH,"
7540 PRINT " EAST,WEST, UP OR DOWN. TYPE P FOR SCORE"
7550 PRINT
7560 PRINT
7570 RETURN
8000 REM -------------------------
8010 REM DESCRIBE THE CURRENT LOCATION
8050 I=INT(N/5)
8060 J=N-5•I+1
8070 REM THERE ARE ENOUGH STATEMENT NUMBERS HERE TO HANDLE NODES
8080 REM ZERO THROUGH 49. YOU WILL HAVE TO ADD MORE IF YOU ADD
8090 REM NODES 50 AND BEYOND.
8100 ON I+1 GO TO 8200,8210,8220,8230,8240,8250,8260,8270,8280,8290
8200 ON J GO TO 9000,9010,9020,9030,9040
8210 ON J GO TO 9050,9060,9070,9080,9090
8220 ON J GO TO 9100,9110,9120,9130,9140
8230 ON J GO TO 9150,9160,9170,9180,9190
8240 ON J GO TO 9200,9210,9220,9230,9240
8250 ON J GO TO 9250,9260,9270,9280,9290
8260 ON J GO TO 9300,9310,9320,9330,9340
8270 ON J GO TO 9350,9360,9370,9380,9390
8280 ON J GO TO 9400,9410,9420,9430,9440
8290 ON J GO TO 9450,9460,9470,9480,9490
8400 IF TC<>N THEN 8500
8410 PRINT
8420 PRINT "THE TREASURE IS HERE!"
8500 IF TC<>T2 THEN 8600
8510 IF T1=T2 THEN 8600
8520 IF T1<>N THEN 8600
8530 PRINT
8540 PRINT "A NOTE ON THE WALL SAYS"
8550 PRINT " 'PIRATES NEVER LEAVE THEIR TREASURE"
8560 PRINT " TWICE IN THE SAME PLACE!"
8600 RETURN
9000 REM -------------------------
9001 REM FIRST DATA STATEMENT IS NUMBER OF NODES, AND THE 2
9002 REM HIDING PLACES FOR THE TREASURE.
9003 DATA 42,23,12
9004 DATA 42,23,12
9005 DATA 1,0,0,0,0,0,0
9006 PRINT "YOU'RE AT A DEAD END!"
9007 GO TO 8400
9010 DATA 2,-2,101,-2,0,0,0
9011 PRINT "YOU CAN'T GO IN THAT DIRECTION"
9012 PRINT
9013 DATA 3,3,2,1,10,106,8
9014 PRINT "A TUNNEL GOES NORTH-SOUTH."
9015 PRINT "THERE IS AN OPENING TO THE WEST."
9016 GO TO 8400
9017 DATA 4,3,30,2,11,2,1
9018 PRINT "YOU'RE ON THE BRINK OF A PIT."
9019 GO TO 8400
9020 DATA 5,8,8,15,10,8,16
9021 PRINT "YOU'RE OUTSIDE THE CAVE."
9022 PRINT "GO SOUTH TO ENTER."
9023 GO TO 8400
9024 DATA 6,16,3,2,10,2,2
9025 PRINT "YOU'RE AT THE HOME OF THE GNOME-KING."
9026 PRINT "FORTUNATELY, HE'S GONE FOR THE DAY"
9027 GO TO 8400
9028 DATA 7,-2,101,2,0,0,0
9029 PRINT
9030 PRINT
DATA 8,18,18,15,10,18,9
9081 PRINT "YOU'RE LOST IN THE WOODS."
9087 GO TO 8400
DATA 9,2,33,5,1,0,-2
9090 GO TO 8400
DATA 10,-2,101,-2,0,0,0
9097 PRINT "YOU'RE NOT GOING TO GET FAR, DIGGING!"
9102 PRINT "THROUGH ROCK."
9103 PRINT
9107 GO TO 8400
DATA 11,1,13,4,2,1,2
9110 PRINT "YOU'RE AT THE BOTTOM OF A PIT. A LITTLE"
9112 PRINT "STREAM FLOWS OVER THE ROCKS HERE."
9117 GO TO 8400
DATA 12,36,2,1,2,1,2
9120 PRINT "YOU'RE AT A DEAD END!"
9127 GO TO 8400
DATA 13,2,37,2,1,11,14
9130 PRINT "YOU'RE IN A FANOT. HIGH ON THE WALL"
9132 PRINT "ABOVE YOU IS SCRATCHED THE MESSAGE"
9134 PRINT "'BILBO WAS HERE!'"
9137 GO TO 8400
DATA 14,13,1,19,2,31,31
9140 PRINT "YOU'RE IN A CANYON. HIGH ON THE WALL"
9142 PRINT "YOU'RE AT THE NORTH SIDE OF A CLIFF."
9147 GO TO 8400
DATA 15,-2,101,-2,0,0,0
9150 PRINT "IT'S A TIGHT SQUEEZE. YOU CAN'T" PRINT
9152 PRINT "GET PAST WITH THE TREASURE."
9153 PRINT
9157 GO TO 8400
DATA 16,5,33,2,10,1,106
9160 PRINT "YOU'RE IN A LONG HALLWAY. A TIGHT TUNNEL"
9162 PRINT "GOES EAST, AND YOU CAN WALK IN THE"
9164 PRINT "SOUTH OR WEST. THERE IS LIGHT!"
9167 GO TO 8400
DATA 17,-2,101,-2,0,0,0
9170 PRINT "IT'S A TIGHT SQUEEZE. YOU CAN'T" PRINT
9172 PRINT "GET PAST WITH THE TREASURE."
9173 PRINT
9177 GO TO 8400
DATA 18,-2,101,8,0,0,0
9180 PRINT "I DON'T THINK YOU CAN FIND THE CAVES."
9187 GO TO 8400
DATA 19,226,2,2,14,1,42
9190 PRINT "YOU'VE AT THE TOP OF A CLIFF."
9192 PRINT "BELOW YOU A MESSAGE SAYS"
9193 PRINT "'BILBO WAS HERE!'
9197 GO TO 8400
DATA 20,226,1,2,2,25,2
9200 PRINT "YOU'RE AT THE NORTH SIDE OF A CLIFF."
9202 PRINT "YOU WERE JUMPING ECHOES FROM"
9203 PRINT "BELOW ARE THE ONLY INDICATION OF DEPTH."
9207 GO TO 8400
DATA 21,1,226,2,2,38,25
9210 PRINT "YOU'RE IN A NARROW DEEP CAVEM."
9212 PRINT "YOU EXAMINE THE SACRED RIVER RUNS"
9214 PRINT "THROUGH CAVERNS MEANINGLESS TO MAN."
9217 GO TO 8400
DATA 22,-2,33,13,50,29,30
9220 PRINT "YOU'RE ON THE LEDGE ABOVE THE GUILLOTINE ROOM."
9227 GO TO 8400
DATA 23,2,1,2,31,2,2
9230 PRINT "YOU'RE AT THE TOP OF A CLIFF."
9231 PRINT "YOU'RE AT THE TOP OF A CLIFF."
9237 GO TO 8400
DATA 24,-2,101,19,0,0,0
9240 PRINT "I HEAR THE GIANT THERE!!!"
9241 PRINT "YOU DON'T HAVE TO GO BACK!"
9243 PRINT
9247 GO TO 8400
DATA 25,21,20,2,2,1,19
9250 PRINT "YOU'VE AT THE GIANT'S CAVERN. BETTER"
9252 PRINT "NOT BE HERE WHEN THE GIANT COMES!"
9257 GO TO 8400
DATA 26,-2,65,-2,50,11,14
9260 PRINT "YOU'RE IN THE QUEST RESEARCH AND"
9261 PRINT "YOU'RE IN THE QUEST RESEARCH AND"

Listing 2 continued on page 186
Listing 2 continued from page 185:

9260 PRINT "OUTSIDE THE CRYSTAL PALACE."
9261 PRINT "YOU'RE IN A PREHISTORIC DWELLING."
9262 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9263 PRINT "OUTSIDE THE CRYSTAL PALACE."
9265 PRINT "YOU'RE IN A PREHISTORIC DWELLING."
9266 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9267 GO TO 8400
9268 DATA 38,2,2,1,116,1,2
9269 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9270 GO TO 8400
9271 DATA 38,2,2,1,116,1,2
9272 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9273 GO TO 8400
9274 DATA 38,2,2,1,116,1,2
9275 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9276 GO TO 8400
9277 DATA 38,2,2,1,116,1,2
9278 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9279 GO TO 8400
9280 DATA 38,2,2,1,116,1,2
9281 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9282 GO TO 8400
9283 DATA 38,2,2,1,116,1,2
9284 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9285 GO TO 8400
9286 DATA 38,2,2,1,116,1,2
9287 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9288 GO TO 8400
9289 DATA 38,2,2,1,116,1,2
9290 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9291 GO TO 8400
9292 DATA 38,2,2,1,116,1,2
9293 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9294 GO TO 8400
9295 DATA 38,2,2,1,116,1,2
9296 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9297 GO TO 8400
9298 DATA 38,2,2,1,116,1,2
9299 PRINT "YOU'RE IN A TWISTY LITTLE MAZE."
9300 GO TO 8400

Text continued from page 179:

modifications on most machines which have enough memory to hold it. The number of locations has been increased slightly, and a few surprises have been added to make it more interesting. The original Quest was made for seventh-graders at Peninsula School, who were doing a map making unit. For this reason, most of the connections between locations work as if they were in physical three-dimensional space, although there is no requirement for this in the program.

The program and the description of the cave are well documented by the comments in the code, so I won't go into great detail here. Besides its description, each location has a set of six numbers which give the next location to move to, in case of a move NORTH, EAST, UP, DOWN, WEST, or SOUTH. Special events, such as the pirate and the treasure, are done in the program rather than in the descriptions. There is a provision in the connection codes for an immediate return to the location you came from, which is used, for instance, at the location called "YOU CAN'T GO IN THAT DIRECTION." There is also provision for different connections chosen according to a random number, and for different connections depending on whether or not you are carrying the treasure, and whether or not the pirate has found you.

A Final Statement

It is possible to get through the cave by reading the program and decoding the data which defines the connections. If you do that, you will deprive yourself of the pleasure of finally finding your way through. It is also possible to "help" a friend by telling him how to get through. I don't think the easy pleasure of knowing how to get through can equal the joy of discovering the way, or the satisfaction of having discovered it, or the excitement of being on the way to discovering it. I also don't think that anyone who merely plays Quest can have as much fun as I have had in writing it, and watching other people use it.

Acknowledgments

My thanks go to Larry Tesler and Phyllis Cole, of the Peninsula School Computer Project, for their encouragement and technical help, and to Mary Artibee for her help with this article. A tape of the PET version of Quest is available for $9.95 from the Peninsula School Computing Project, Menlo Park CA 94025.
As you might surmise from the title, this textbook promotes structured programming, design, and testing concepts. Although familiarity with current hardware, systems, or software doctrines is not a prerequisite for reader comprehension, the greatest benefits will accrue to those individuals active in software development or maintenance efforts.

The book's tone and structure are established in the first chapter with a discussion of the seven most desirable qualities of a good program. The program should work well and according to specification; the simplicity of design should reduce development, testing, and maintenance costs; and the program's inherent flexibility should allow change through expansion, modification, or upgrade.

Of perhaps more importance to the average programmer is the analysis of ten practices which commonly exacerbate the debugging problem. Also included are suggestions for easing the difficulties of maintaining or modifying existing programs for the maintenance programmer.

The next chapter deals with top-down design, coding, and testing. Top-down design is the process by which the programmer identifies the major functions of a programming problem, and organizes the solution in such a manner that it is recognizable to both the computer and the maintenance programmer.

Mr. Yourdon provides five suggestions for successfully applying this concept. The benefits and disadvantages of flowcharts figure prominently in the arguments for and against the concept of writing code concurrently with designing the top-down structure. An offshoot of this discussion of top-down coding concerns how best to display modular code: the horizontal and vertical approaches have unique merits which should be considered prior to presenting the code.

Following the presentation of the nature and advantages of top-down testing, the author deals with practical variations of the pure approach to top-down design, coding, and testing. These modifications are frequently the result of organizational problems and management compromises. Two of the most successful variations discussed are the IBM developed structured walkthrough technique and IBM's highly innovative chief programmer team concept.

In Chapter Three, the reader is introduced to the concept of modular programming (precursor to the now popular structured programming technique). After a discussion of the characteristics of a programming module, the pros and cons of modularity are detailed. Techniques for achieving modular programs are of particular interest, especially those sections dealing with decision tables, separate I/O (input/output) functions, and use of symbolic parameters. The chapter concludes with the closely related subject of general purpose subroutines.

After a review of the history and background of the structured programming movement, both the theoretical and practical aspects of the concept are detailed in Chapter Four. While analyzing structured programming's objectives and motivations (in terms of testing, productivity, clarity, and efficiency), Mr. Yourdon highlights

---

**At last... the mechanical interface!**

Turn your electric typewriter into a low cost, high quality hard copy printer.

User list $395.00

The new I/O Pak from Rochester Data, Inc. interfaces the keyboard of any commercially available electric typewriter with any computer. The result: low cost, high quality hard copy. Write today for more information.

3100 Monroe Avenue, Rochester, New York 14618

Circle 326 on inquiry card.
There is a reason so many Datec customers are communication line technicians, computer engineers and data processing experts. They recommend Datec for the simple, solid state, rugged design that has proven to be more reliable than bigger name couplers.

Acoustic couplers are the work horses of a data communications system. They should work every time, every day, so that you can install them and then forget them. No wonder communication line technicians have nicknamed the Datec 32 the "heavy duty" acoustic coupler.

CRYSTAL CONTROL
Crystal controlled trans­mitter and receiver assures long-term stability and reliability, even in the most unfriendly operating environment.

BIG EARS
Rugged, tight-fitting rubber cups effectively seal out vibration and room noise. They aren’t beautiful, but they work.

BURNEEDAN
Datec may be the only company that tortures every PC board by burning it each one for 168 hours at elevated temperatures.

DATEC INC., PO BOX 839, CHAPEL HILL, NC 27514, (919) 967-5605

DATEC DID IT! Quality — designed in, burned-in, and guaranteed.

Circle 98 on inquiry card.

significant differences between structured and modular approaches to program design. Although structured programming is becoming the standard method of achieving modular programs, the concept involves more than the simple conjunction of "GOTO-less" programming with the process, loop, and binary-decision constructs. As with most evolving techniques, there are tradeoffs in terms of efficiency, convenience, hardware and software requirements which concern many of those involved with computer science. The structured technique is not a panacea for programming ills.

Mr Yourdon has found that imposing structure upon poorly designed, unstructured programs usually results in poorly designed, GOTO-less, structured programs that are almost as difficult to understand and debug as the originals. As a result, he recommends that experienced programmers rework their previously unstructured code only as a means of bringing their thought processes into compliance with current top-down design concepts. There are three common techniques for restructuring this code, each with its own strengths and weaknesses: the duplication-of-coding technique which is recommended for programs with network or lattice structures; the state-variable technique which is helpful to the maintenance programmer and can be applied to very complex processing programs; and the Boolean flag technique which is applicable to loop-oriented programs.

The chapter concludes with a discussion of the applicability of structured programming to the currently available high level programming, systems implementation, and assembly languages. ALGOL, PL/I, COBOL, FORTRAN and assembly languages are covered quite extensively; PL/S, BLISS, PL/360, Burroughs B5500 ESPOI, Project MAC EPL, and GE-645 MULTICS languages are discussed in less detail. I noted with interest the author’s claim that FORTRAN and assembly languages do not provide the necessary facilities for structured programming.

In Chapter Five, programming method, system software, and system hardware figure prominently in the discussion of the elements of programming style which dramatically affect the coding and debugging process. Of particular note are the questions concerning how to best reduce program complexity so as to limit the introduction of errors, and how to construct programs so that someone other than the original programmer can read and comprehend the logic flow.

The focal point of the next chapter is defensive programming — the practice of writing programs in such a way that the
inevitable bugs are readily noticeable to both programmer and user. Effective arguments are presented against the most common objections to this antibugging technique. Following the presentation of a comprehensive list of aspects of a computer program which require checking, the author summarizes nine useful error-checking techniques. Although many of the antibugging techniques belong more properly in the realm of system, rather than program design, the discussion is valid at both the programming and coding levels.

Testing is reviewed in Chapter Seven. The magnitude of this effort can reach staggering proportions on projects which utilize many programmers, large numbers of man-hours, and vast segments of code. Current module, system, and acceptance testing techniques expose most common types of error. These errors are divided into eight categories: logic, documentation, overload, timing, throughput and capacity, fallback and recovery, system hardware/software, and standards errors.

It is Mr Yourdon's contention that a program is economical to test and debug only if the team programming concept is used in combination with an antibugging programming style and an unambiguous, structured language. Also, because programmers use slow, tedious methods and are psychologically inclined to justify their output as correct, he advocates the removal of the human element from the testing process. Ideally, the testing process would be fully automated: the automatic test harness would collect the test input from the automatic test data generator, pass it to the program being tested, and use an automatic output checker to list the discrepancies. Upon program execution, the automatic testing monitor would print a report showing the portions of the program exercised by the test data. The program would be subjected to a thorough retesting if it is further modified. The chapter concludes with a review of some of the experimental techniques currently being evaluated for their abilities to increase program reliability.

Once the existence of a bug has been established, it is time to employ the technique of finding and correcting the error, known as debugging. The procedure is more art than science, and in Chapter Eight the reader is provided with both an explanation for 26 common bugs and 11 detailed suggestions to assist in the formulation of a workable debugging strategy. A terminal-oriented debugging system greatly reduces the frustration of using memory dumps and program traces. The system is called DDT or Dynamic Debugging Technique. After describing the general features of the process, the author expands his explanation on the currently available stand alone and sophisticated timesharing and realtime packages. The chapter concludes with the implementation of a simple version of DDT.

The four programming exercises in the appendices illustrate the principles of program design and structured programming, and allow the reader to apply the techniques presented in the book to real problems. The two management system problems are quite complex and can best be attacked by a team of three to four programmers. The master file update and tic-tac-toe problems are more simple and are suitable for individual programmers.

This highly informative book is packed with information for both the commercial and personal aspects of programming. The book's heavy concentration on the general philosophy and techniques of good programming deserves the highest accolades. Techniques of Program Structure and Design should be a part of every dedicated computer user's library.
This book provides a good introduction to the top-down structured programming concept. This concept requires, as a first step, that a problem's functional specifications be determined. The problem is then refined until that programming code directly expresses the resulting subfunctions, or modules. The authors believe that programmers who use the top-down structured methodology are more productive, and consistently produce a reliable product which works despite continuous testing or future modification. The book also details recent programming advances in the large system, multiprocessing environment.

Chapter 1 presents an overview of structured programming with an emphasis on the development of a language-independent methodology whose principal goal is program reliability. Of special interest is the section on the importance of developmental and modification software (in terms of time, cost, and reliability). Following a preview of the remaining chapters, the authors discuss their reason for illustrating programming concepts with code fragments in the PL/I language.

Chapter 2 emphasizes program reliability. The data processing industry spends a great amount of time and money to insure that programs are developed with as few flaws as possible. The current doctrine of testing programs for reliability is gradually giving way to a more efficient method of checking for correctness. Finally, the DO-WHILE construct is introduced by emphasizing the importance of properly initializing conditional loops, and explicit guidelines for programming conditional iterations are given.

In the next chapter, the SEQUENCE and IF-THEN-ELSE programming tools are discussed. Correctness questions, which the programmer should pose while using these constructs, are used as an aid in determining the functional specifications of the program. Program specifications must be determined prior to commencing the actual coding process because the authors consider inaccurate specifications to be the most important source of software errors. The ITERATIVE-DO, SELECT-CASE, REPEAT-UNTIL, and LOOP-EXITIF-ENDLOOP constructs are formed by combining the basic figures.

Unfortunately, the authors elect to leave the related correctness questions as an exercise for the reader, feeling that "...the reader is now sufficiently aware of the correctness considerations to properly examine his use of the nonbasic figure." Although the authors admit to writing for the computer professional, rather than the mathematician or newcomer to the computer field, I feel they did their readers a disservice. These correctness questions are vital (without them, the book suffers from a lack of comprehensiveness which limits its usefulness to both the salaried professional and computer hobbyist). Chapter 3 concludes with an interesting section on how to impose structure on programs written in FORTRAN and COBOL. The examples are illuminating.

The advantages of the top-down structured approach to program design, coding, and integration are discussed in Chapter 4. Top-down segmented implementation is compared (somewhat one-sidedly) with the more standard bottom-up strategy. Among the advantages discussed is the following: the effectiveness of top-down structured programming in eliminating construct errors (missing paths, inappropriate path selection, and inappropriate action under a given condition). The top-down method concentrates on the more time-consuming and error prone aspects of programming during the initial, rather than the final stages of development. A structured program has the advantage of becoming its own principal documentation. And the final advantage, especially associated with large systems pro-
programs, is that integration and debugging are easier to implement under the top-down strategy. The programming technique of recursion is introduced in the concluding sections of the chapter.

The remaining two chapters and two appendices are of less interest to the computer hobbyist. Chapter 5 examines IBM's innovative "chief programmer team" organization and its managerial approach to the operation of software projects. Chapter 6 provides extended examples of the top-down structured approach to program design, with the primary focus on parallel processes (whereby several computations proceed either simultaneously on separate processors, or by multiprogramming on a single processor) and multiprocessor resource management. The chapter concludes with the specification and top-down design in pseudocode, of a hypothetical multiprogrammed, multiprocessor operating system. The REPEAT-UNTIL, SELECT-CASE, and LOOP-EXIT-ENDLOOP structured modules are implemented in PL/I in Appendix A. In Appendix B, six of the seven structured programming constructs are specified in 360/370 assembly language macros.

As a result of the omission mentioned earlier, I have qualms about recommending this book as a mandatory reference resource. If you are interested in using structured techniques to increase your programming productivity and reliability, I would recommend that you get a copy of the book through a nearby college or computer club library and spend a few hours reading the material. I found the authors' style lucid, although on occasion I had difficulty following some of their examples due to my unfamiliarity with the PL/I language.

Overall, the book successfully presents the authors' contention that structured coding, top-down design, formal and informal proofs of correctness, chief programmer team organizations, and code reading result in reliable, flexible software. I fully concur with their belief that "...top-down structured programming is one way to make programming the enjoyable activity it should be."

Lee C Matthews
9659B Jefferson
Ellsworth AFB SD 57706

---

**A new star on the Horizon**

**MMike's**

**CSUB (Common SUBroutines)**

Developed for North Star DOS and BASIC by Micro Mike's, Incorporated

CSUB is a multi-dimensional programming package. Among benefits CSUB offers both user and programmer are:

- Non-destructive cursor positioning
- Automatic display of error messages or bulletins
- Strict control of all data passing between CRT and CPU, including:
  - Complete parameter checking of all numeric data input
  - Complete formatting of all numeric and alphanumeric data displayed or input
  - Automatic mask and data display
  - Automatic date input and display (automatic display of slashes between month, day, and year)
- Complete numeric control on single character alphanumeric inputs
- User's ability to "back up" to last logical input
- Strict control of data passing between external storage devices (i.e. disk drives) and CPU, including:
  - Automatic file OPENing and CLOSEing for most file accessing
  - Sequential file accessing
  - Random file accessing with automatic calculations of file vectors
  - Keyed file accessing with virtually no limits on number of keyfiles
  - Automatic sorting of keyfiles.

**Micro Mike's, Incorporated**

**PROGRAM LIBRARY**

A one-time fee of $500 (soon to be increased) buys for the purchaser continuous unlimited access to the programs, updates and revisions in the PROGRAM LIBRARY:

| CSUB disk and documentation package | $49.95 |
| 4SHARE (timesharing for Horizon, Release 4) | $49.95 |
| 5SHARE (timesharing for Horizon, Release 5) | $49.95 |
| DOSCHG4 (8" disk drive interface patch to Release 4 North Star DOS and BASIC) | $49.95 |
| DOSCHG5 (8" disk drive interface patch to Release 5 North Star DOS and BASIC) | $49.95 |

**Making technology uncomplicated**

905 South Buchanan
Amarillo, Texas 79101
806-372-3633

July 1979 © BYTE Publications Inc 191
BASIC Bit Twiddling

Ralph Owen
POB 202
Enterprise KS 67441

The following are several functions which allow the user to manipulate individual bits and nybbles (groups of four bits) using BASIC. I like to use these functions with the PEEK and POKE commands.

Nybble functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Formula</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNU(N)</td>
<td>INT(A/16)</td>
<td>Reads upper nybble.</td>
</tr>
<tr>
<td>FNL(N)</td>
<td>A - 16*FNU(N)</td>
<td>Reads lower nybble.</td>
</tr>
<tr>
<td>FNA(N)</td>
<td>16*N + FNL(N)</td>
<td>Sets upper nybble to value of N.</td>
</tr>
<tr>
<td>FNB(N)</td>
<td>FNU(N) + N</td>
<td>Sets lower nybble to value of N.</td>
</tr>
</tbody>
</table>

Bit functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Formula</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNQ(N)</td>
<td>INT(A/2^N) - 2*INT(A/2^(N + 1))</td>
<td>Reads Nth bit.</td>
</tr>
<tr>
<td>FNR(N)</td>
<td>A - FNQ(N)*2^N</td>
<td>Resets Nth bit to zero.</td>
</tr>
<tr>
<td>FNS(N)</td>
<td>A + (1 - FNQ(N))*2^N</td>
<td>Sets Nth bit to one.</td>
</tr>
<tr>
<td>FNT(N)</td>
<td>FNR(N) + FNS(N) - A</td>
<td>Toggles the Nth bit.</td>
</tr>
</tbody>
</table>

A contains the value of the byte. N contains the value of the nybble.

These functions can be used to manipulate individual bits and nybbles in a byte.

Here are two examples of the use of these functions:

1000 REM FIND LOWER NYBBLE VALUE
1010 A = PEEK(16422)  
1020 V = FNL(A)  
1030 PRINT "VALUE OF LOWER NYBBLE IS"; V  
9999 END

2010 A = PEEK(16422)  
2020 REM SET EIGHTH BIT OF DECIMAL  
2030 REM LOCATION 16422 TO ONE  
2050 A = FNS(8)  
2060 POKE 16422, A  
9999 END

SIMULATION IS a collection of the best articles from BYTE Magazine on this useful computer technique, plus exciting new material on the subject. For those who view the personal computer user as lacking the ability to write and maintain simulation programs, but also actual programs for simulating many situations. Theoretical and practical applications of this technique are explored in articles dealing with specific aspects of simulation. Learn how to find solutions to your problems through SIMULATION.


$6.00

Buy this book at your favorite computer bookstore or order direct from BYTE BOOKS. Add $5.00 per book for postage and handling.
**World Power Systems: A Report**

This spring, World Power Systems Inc of Tucson AZ began "business" with a promotion campaign, ordering goods from suppliers, and soliciting orders and money from customers. It is now known that only a few initial orders were filled (apparently to establish credibility), and that money accompanying later orders was simply stashed away.

Because of this, our industry has received a black eye, and many of us, BYTE included, are apparently out of substantial amounts of money. Acting on a tip received April 25, the Pima County Attorney's Office promptly investigated and, within a few days, had sufficient evidence of illegal activities to obtain warrants for the arrest of those responsible for the operation. An arrest was made, goods were seized and impounded, and others, who were out of Tucson at the time, are being sought by the authorities.

The question that all of us must be asking is "How can we prevent this from happening again?"

It would be nice if there were a simple answer. The main things consumers can do are (1) don't send money unless you have reason for confidence in the company and (2) promptly report (eg: to postal authorities) anything suspicious.

What neither consumers nor the media can afford to do, however, is to establish a presumption of guilt. Just because a company is new does not mean that the media can refuse advertising or that consumers should avoid the company's products. Such actions would only lead to stagnation in the industry and possible claims of unfair competition and/or antitrust violations. Indeed, since many of the biggest companies in the industry began in founders' homes, such a policy might have retarded the development of the personal computer industry by several years.

BYTE has had, and continues to have, stringent standards on the advertising it accepts. Advertisements have been rejected for such reasons as being tasteless, misleading, or having originated from unreliable companies.

As to reliability, we have to depend in part on readers' input. Financial reports on advertisers are, unfortunately, of limited value. Some financially unstable companies are totally honest, while others, with a superficially impressive report, may turn out to be fraudulent.

At BYTE, any reader's complaints about advertisers are sent to the advertiser along with our request that BYTE be notified of what action is being taken. Since there are always two sides to every argument, having a single disgruntled customer would be an adequate reason for sanctions against the advertiser. A series of complaints without adequate response by the company, however, is cause for BYTE refusing to run further advertisements by that company.

In summary, there is no absolute protection in any industry against well-planned and well-executed frauds. Prompt communication between (1) consumers, (2) reputable companies, (3) the media, and (4) the authorities can nip poorly planned frauds in the bud and limit the damage that is done.

Chris Morgan
BYTE Executive Editor
As noted in a recent editorial, I am in the process of designing and building a new computer system based on the 6809 processor. It is my intention, as this design evolves, to provide a fairly complete set of plans in the form of an irregular series of articles in BYTE. The building and developing of the software systems of homebrew computers is, after all, the basis from which personal computing has developed. This series of articles will document the development of an up-to-date design that utilizes contemporary components which were unavailable to me when I first started building crude and imperfect homebrew computers in 1974.

The regularity of this documentation will be dependent upon the time that is available to engineer, build and test the component parts. I will try to provide an update on the progress of the project with each future issue of BYTE. From time to time there will be gaps in the series since, like all people, I have only 24 hours each day in which to work. Having issued this caveat concerning the irregularity of the information, let us turn to the starting point of my documentation; a physical basis for a bus oriented homebrew computer.

The following is documentation of a key part of any homebrew computer; its backplane. At this stage, the computer system is depicted in photographs 1 through 5, which I took while assembling the backplane on a recent evening. For the photographers among our readers, all these pictures were taken with a 35mm single lens reflex (SLR) camera, highly stopped down (f/32, f/22, depending on lens) to emphasize depth of field, and using lots of light (1200 W).

Photo 1: This pile of parts represents the beginning of the project’s physical hardware assembly. At left is a set of eight copper rods made from #12 gauge household electrical wire. After stripping the insulation, one end of the wire is clamped in my bench vise, and the other end is clamped in “Vise-Grip” pliers. Five uniform, careful rotations of the wire while holding tension suffice to make the rods as straight as those shown in the photo. This torsion straightening process gives amazing results. These rods will be the bus wires of the power and ground distribution.

The matrix for assembly of the backplane is one Vector Electronic Co #3719-4 “P” pattern prototyping board. This board was chosen as standard for the new computer because it has a 0.1 inch (.254 mm) square grid consistent with integrated circuit sockets, and an identical 0.1 inch spacing for the edge connector sockets. Thus, the same board style that will be used for the assembly of the computer modules can also be used for assembly of this motherboard.

The final part that is going into the physical assembly of the backplane is a set of 6 edge connector sockets for the circuit modules. The sockets have 72 circuit connections in two rows of 36 pins.
HOW TO BUY A BUSINESS COMPUTER

WHEN TO BUY • WHEN NOT TO BUY • HOW TO CHOOSE CONSULTANTS • DATA GATHERING • PACKAGED VS CUSTOM SOFTWARE • CONTRACTS • FINANCING • WARRANTIES AND MAINTENANCE • INSTALLATION MANAGEMENT AND MUCH MORE . . . .

"SUPER BOOK!" DP Consultant, Mercer Island, Washington. "I wish I'd had this book before I bought my computer!" TV Station General Manager, California.

"It's the best book I've seen on the subject. Send me 5 copies to send to my clients!" CPA, Orange County, California, "I really learned something from this book and I'm in the business. I'm recommending it to my clients." DP Consultant, Alhambra, California.

Here's a straight forward approach to business computer purchase and installation. It will save you time and money. ORDER NOW! If not completely satisfied, return within 30 days for a full and immediate refund.

Rush . . . . copies of "Winning The Computer Game" at $12.95 per copy. (CA residents add 6% sales tax)

GATHERING • PACKAGED VS CUSTOM SOFTWARE • CONTRACTS • FINANCING HOW TO CHOOSE CONSULTANTS • DATA purchase and installation. It will save you time and money. ORDER NOW! If not completely satisfied, return within 30 days for a full and immediate refund.

The Independent Newsletter of Heath Co. Computers

Should it be a Heathkit®?

Whether you are considering the purchase of an additional peripheral or your first computer you should know more about Heathkit® computer products. Heathkit® has a continuing commitment to selling well-documented computer kits and software. Are they for you? How can you find out?

Read Buss: The Independent Newsletter of Heath Co. Computers, where information on new products is printed as it leaks out of Benton Harbor, not held back to suit a marketing plan. Buss is not a company-controlled publication, so it can deal with weaknesses of Heathkit® products as well as their strengths. It features news of compatible hardware and software from other vendors. Every Buss issue has candid accounts of experiences of Heathkit® owners. Results of their discoveries, which often include hardware modifications, save subscribers headaches—and money. That's proven by two years' experience. So Buss can guarantee a full refund any time you're not satisfied.

Buss is mailed first class (by airmail outside North America). The 24-issue subscription gives you the choice of starting with the latest issue or with available back issues (about 8 are still in stock). Send $8.00 for 12 issues or $15.25 for 24 (overseas, $10.00/12; $19.25/24 US funds) to: Buss

325-B Pennsylvania Ave., S.E.
Washington, DC 20003

TINY' PASCAL

for TRS-80® & NORTH STAR®

Now you too can have Pascal! The Chung/Yuen 'Tiny' Pascal has been specially designed for TRS-80 & North Star owners. The full power & elegance of 'Tiny' Pascal is at your command. Programs written in 'Tiny' Pascal run at least 4 times faster than the same program in BASIC! 'Tiny' Pascal is also a great way to learn Pascal Programming, & fun too.

The minimum system requirements are: Level II, 16K for TRS-80, & 24K for North Star (specify single or double density).

But most important, you also get source to 'Tiny' Pascal written in Pascal with each purchase! You can even compile the compiler! (Requires 36K for North Star systems, & 32K, Level II for TRS-80). You can customize your own version, or just use it the way it is.

'Tiny' Pascal is a subset of Standard Pascal & includes:

RECURSIVE PROCEDURE/FUNCTION, IF-THEN-ELSE, REPEAT/UNTIL, 'PEEK & POKE', WHILE, CASE, & MORE!

Also you can save & load programs.

You get all this & more, plus a user's manual for $40.00.

available from:

SUPERSOFT

P.O. Box 1628
Champaign, IL 61820
(217) 344-7596

All orders pre-paid, Illinois residents add 5% sales tax
Photo 2: The physical assembly of the backplane begins by noting the fact that a single #12 gauge wire will fit quite nicely between a set of four pins on a 0.1 inch grid. Thus, if we insert all the sockets in the backplane board, we can drop one copper bus connection across four corresponding pins of all six sockets. The 12 gauge wire size for the buses is just slightly larger than the 0.06 inch spacing between the wire wrap socket pins of a 0.1 inch grid. This fact causes the pins to spread just a bit, locking the sockets in place prior to soldering. In this photo we show eight bus wires occupying 32 pins of the backplane socket.

Looking ahead to the design of my computer and basing my conclusion on previous experience, I will need a total of four power supply buses. A standard +5 V is the main power supply, used by most of the digital integrated circuits. Ground is the common point for all power supplies. Two symmetrical supplies of −12 V and +12 V will be used by analog circuits.

Noting that a symmetrical arrangement of the backplane prevents power supply destruction through inadvertent reversal of boards, two sets of four buses are used. As we will see later (photo 4), the innermost buses will become the basis for the ground distribution grid.

With four power voltages occupying a total of 32 pins, the 72 pin sockets of the backplane have 40 uncommitted pins available for communications between boards. This is more than adequate for a good general purpose computer system based on an 8 bit microprocessor, such as the 6809 I will be using with this design.

Photo 3: Assembly begins with the outermost bus, laying it down in its niche in the pin forest of all six sockets. It is then soldered to each set of four pins, as depicted in this photo. I used a 120 W light-duty soldering gun for this operation, since the extremely high heat carrying capacity of the copper wires made my 25 W soldering iron impossible to use.

After each bus wire is soldered in place, the four pins at each socket are clipped off just above the copper wire and solder bead. This process is repeated for each of the eight bus wires assigned to the power voltages of the new computer.

Care must be taken while soldering to avoid forming a bridge between adjacent buses. The last step in soldering a bus wire is incomplete in this photo: the wire is just resting on the four holes in the tab from the backplane’s edge connector. When soldering this part, be extremely careful about forming bridges from one bus wire to the next.
Photo 4: The completed backplane assembly includes a set of jumpers bridging the symmetrically arranged power supply voltages. Six bus wire jumpers connect the inner ground buses across the socket area. Three heavy insulated stranded wire jumpers create an aesthetically pleasing, but electrically useless arch form at one end of the board.

The bus wires in this photo of the assembly process have been soldered to the edge connector pins corresponding to power distribution voltages. Not shown at this stage is a set of bypass capacitors installed between the three voltages and ground. On each voltage, six 0.1 μF ceramic capacitors and one 3.3 μF tantalum electrolytic were installed for local bypass of the power supply voltages.

Photo 5: No backplane idea is complete without a discussion of the physical support of the boards in the final assembly. Here we show the newly completed backplane together with six boards and a set of 1.375 inch (1 1/8 inch) spacers.

Readers wishing to duplicate this board should learn from an imperfection I introduced. If the boards in this picture appear to be a little crooked, rest assured that this is true, and not caused by the wide angle lens. The actual center to center spacings I finally used (count the holes in photo 4 if you wish) were 1.4, 1.5, 1.7, 1.4, and 1.5 inches! The spacing between backplane sockets should be 1.4 inches, center to center. In my final assembly, I will have to use extra washers as spacers due to this flagrant indiscretion during initial assembly.

Physically, this completes the motherboard and its power supply electrical connections. In part II of this description of the physical assembly of my new computer system we will complete the wiring of the backplane (by using a homebrew adaptation of a Vector “Slit-N-Wrap” tool to my electric eraser), wooden cabinetry which forms a base for the computer while hiding its power supply modules, and the final assembly of the computer system's basic hardware.
Mouse

A Language for Microcomputers

This article describes Mouse, a computer programming language which can be implemented on most microcomputers. The word Mouse is not an acronym, merely an appropriate description for something small and active.

There are many available languages for microcomputers already, so the introduction of a new language warrants some explanation. The justification for Mouse is that it incorporates many features of high level languages, yet it can be implemented without the resources needed by most high level languages. More specifically, Mouse programs demonstrate the use and implementation of arrays, functions, procedures, nested control structures, local variables, recursion, and several methods of passing parameters from one procedure to another. Mouse also embodies some of the principles of structured programming, in that it uses nested, single entry control structures, and does not allow unrestricted jumps. Nonetheless, Mouse can be implemented on a minimal system consisting of a microprocessor, 4 K bytes of memory, and a terminal.

All of these features cannot be incorporated into a simple language without making some sacrifices. Identifiers in Mouse consist of a single letter, so that a symbol table is not required. Expressions are written in postfix notation, which is more easily interpreted by a computer than the conventional infix notation. Parameters are passed to subroutines as strings, eliminating the need for complex parameter transmission mechanisms. Mouse programs are easier to write than to read, and it is possible to write Mouse programs that are very obscure. Although readability is a highly desirable feature of a programming language, a language cannot be condemned solely on the grounds that obscure programs can result: witness the popularity of APL. The extraordinary thing about Mouse is that so much can be achieved with such a small amount of implementation effort.

Mouse is descended from an older programming language called Musys. In 1970 I was working for Electronic Music Studios (London) Limited, the company which now manufactures the SYNTHI series of electronic music equipment. At that time the company, under the direction of Peter Zinovieff, was exploring the possibilities of using minicomputers to control electronic music instruments. The studio had two DEC PDP-8 computers, but very little software. My job was to write software which would relieve composers of the tedious chore of entering musical compositions in the form of strings of octal numbers. Since the older PDP-8 was already connected to a variety of digitally controlled oscillators, filters, envelope shapers, and other musical equipment, we decided to use the newer and faster PDP-8 to do the language processing. Space was very limited: the PDP-8 had only 4 K 12 bit words of memory, and a very restricted instruction set by comparison with today's microprocessors.

The system that I designed for the studio enables composers to write their compositions in a high level language called Musys. The heart of Musys is a simple yet powerful macroprocessor. A Musys composition consists of a hierarchy of macroinstructions, in which the higher level macroinstructions determine the overall form of the composition, and the low level macroinstructions specify details such as the pitch and duration of the individual sounds. The Musys interpreter contained about 700 instructions. About 600 additional instructions were required for supporting software, including disk control. The system is described in reference 1. The idea of using a macroprocessor as the basis for a minicomputer
language occurred to me after I had read a paper by the late Christopher Strachey (reference 2).

The remainder of this article consists of two main sections: the first describes the language Mouse; the second describes the implementation of a Mouse interpreter.

**Mouse User's Guide**

A Mouse program is a string of characters. Blanks may be inserted anywhere in the program, but they are ignored by the interpreter, except in a few contexts which are defined below. The last two characters of a program are always $$$. The interpreter starts executing at the first character of the program and processes one character at a time until it encounters the character $$, at which point it stops. This processing sequence is broken only by specific control strings: conditions, loops, macro calls, and formal parameters. These are described below.

When the interpreter encounters the quote character ", it prints or displays characters up to, but not including, the next quote character. For example, the program:

```
"JACQUELINE" $$
```

will print the message:

**JACQUELINE**

The quoted string may contain blanks, which are printed, and the exclamation point ! that prints a carriage return/line feed. Thus the program:

```
"FIRST LINE!SECOND LINE" $$
```

will print:

**FIRST LINE**

**SECOND LINE**

Mouse performs calculations using a stack. An operand pushes a value onto the stack, and an operator removes one or two values from the stack and may replace them with other values. The question mark ? is an operand which tells the interpreter to read a number from the keyboard or input file and push it onto the stack. The operator / removes the top value from the stack and prints it. (Do not confuse the use of / as an operator with its use within quotes.) The following program reads one number from the keyboard and prints it:

```
? ! $$
```

The stack is a last in, first out data structure. The program:

```
??!! $$
```

reads three numbers and prints them in reverse order. For example, if the input file contained:

```
45 46 47
```

then the program would print:

```
474645
```

If we wanted them to be printed in the order in which they were read, we would have to write:

```
?!?!?! $$
```

This program, using the same data as before, would print:

```
454647
```

A decimal integer is another kind of operand. When the interpreter reads an integer, it pushes its value onto the stack. This program prints 365:

```
365! $$
```

We can push two numbers onto the stack by writing them one after the other, with a blank in between. This program prints 7 5:

```
5 7 "" ! $$
```

This is the only context in Mouse where a blank character is required; without it, 5 7 would be read as the single number 57. However, we often insert blanks into Mouse programs to improve their readability. Furthermore, Mouse does not print leading or trailing blanks when it prints a number, so we must include a blank string " " between print operators if we want numbers separated in the output. Note also that Mouse does not process floating point numbers: all operands have integer values.

Mouse has 26 variables, the names of which are A, B, C, .. Z. The name of a variable is an operand, and when the interpreter encounters a variable name, it pushes the address of that variable onto the stack. This program prints the addresses of A and T:

```
A! T! $$
```

The operator . (period) replaces an address on the stack by the value stored at that address.
BUSINESS SOFTWARE

- **INTEGRATED FINANCIAL PROGRAMS**

  All programs integrate by posting financial data to the general ledger journals. They are self-guided and easy to use. You don’t have to be a CPA!

  - **GENERAL LEDGER**
  - **ACCOUNTS RECEIVABLE**
  - **ACCOUNTS PAYABLE**
  - **INVENTORY CONTROL**
  - **PAYROLL**

  $500 Each Program
  $8 Manual Only

- **COMPLETE LEGAL BILLING SYSTEM with ACCOUNTS RECEIVABLE**

  A powerful and flexible system that will produce clients statements with a wide choice of layouts and billing options.

  $1,995 Complete System
  $12 Manual Only

- **MIDAS™ MINI DATABASE SYSTEM**

  Design your own file layouts and change them easily with MIDAS. Complete update, query, tally, sort, report, and reformat capabilities.

  $250 Complete System
  $8 Manual Only

† Over 100 Systems in the field now!
† Immediate delivery on all programs.
† Requirements: CBASIC, dual disk, and 48K memory

*CP/M is a trademark of Digital Research

---

address. If we want to print the value of A we write A./.

The assignment operator is =. An assignment is performed in three steps:

- An address is pushed onto the stack;
- A value is pushed onto the stack;
- The assignment operator is applied.

Thus in order to give the value 17 to X we write:

\[ X := 17 \]

This statement has the effect that \( X := 17 \) would have in a Pascal program. Similarly, we write \( X \cdot Y := \) to assign the value of Y to X in Mouse. The most common programming error in Mouse is to forget the period; no error will be reported, but the calculation will use the address rather than the value of a variable, and the result will probably be wrong.

The arithmetic operators in Mouse are \(+\), \(-\), \(*\), and \(^/\), denoting addition, subtraction, multiplication, and division, respectively. Because operands in Mouse have integer values, \(^/\) means divide and truncate. A Mouse operator is always written after its operands. The resulting notation is called postfix notation or reversed Polish notation, in contrast to the conventional algebraic notation which is called infix notation. Postfix notation may be confusing at first, but it does have some advantages. The infix expression \( A+B \) is written in Mouse as \( A.B.+ \); the periods signify that we are adding values, not addresses. One of the advantages of postfix notation is that parentheses or brackets are not required. The infix expression \( A+B \cdot C \) in which the multiplication is performed before the addition, is written \( A.B.+C. \cdot \) in Mouse. The infix expression \( (A+B) \cdot C \), in which the parentheses indicate that the addition is to be performed first, is written \( A.B.+C. \cdot \).

It is not hard to translate expressions into postfix notation if you remember these two rules:

- Operands appear in the same order in both expressions;
- Operators are written as soon as both operands have been written in full.

As an example, consider the conversion to postfix form of the infix expression \( (A+B)/(C-D) \). First write down the operands in sequence:

\[
A \quad B \quad C \quad D
\]
The addition is performed after \( B \), and the subtraction after \( D \). The division cannot be performed until both of its operands have been computed, so the complete expression in postfix notation is:

\[
A.B. + C.D. - / \]

The arithmetic operators always have two operands in Mouse. The infix expression \(-X\) means \( 0 - X \), and it must be written in the form \( 0.X. - \).

The top value in the stack may be used as an *anonymous variable*. In fact, we have already used the stack in this way, in the program \( \text{? ! $\&$}. \) Here is a more subtle use of this feature:

\[
A \text{ AS=} . A6= ! $\&$ 
\]

These five steps have the following effect:

- \( A \) puts the address of \( A \) on the stack;
- \( A5= \) assigns the value 5 to \( A \) (this uses the stack, but leaves it unchanged);
- \( . \) converts the address of \( A \) to the value of \( A \), which is 5;
- \( A6= \) changes the value of \( A \) to 6;
- \( ! \) prints the value, 5, from the stack.

Care must be taken, of course. If you are writing programs in this fashion, you must at all times know what is supposed to be in the stack. The following example uses the stack to interchange the values of two variables, \( X \) and \( Y \). In most languages this interchange can only be done with a temporary variable. For example, in Pascal we would write:

\[
T := X; X := Y; Y := T 
\]

In Mouse we can write:

\[
X \ Y \ Y \ X \ = = 
\]

The addresses of \( A, B, C, \ldots Z \) are 1, 2, 3, \ldots 26. This means that \( B \), for instance, can be regarded as either a variable in its own right, or as the second element of the array \( A \). The address of \( B \) can be written as either \( B \) or \( A1+ \), and its value as \( B \) or \( A1+ \). A general element of the array \( A \), written \( A[I] \) in Pascal, can be written as \( A1.+ \) in Mouse, for \( 0 \leq I \leq 25 \). Any letter can be used in this way: thus \( K5+ \) is equivalent to \( P \), and \( Z2- \) is equivalent to \( W \) . You must be careful not to use the same address for two different purposes. If you decide to use \( A \) as an array with ten components, you cannot use the variables \( B, C, \ldots J \) in the
Write faster in BASIC, FORTRAN, or COBOL
Document & modify more easily, too

Human-engineered to do the job better. Yes, you really can get flawlessly code faster, using the Stirling/Bekdorf™ system of software development tools with structured programming concepts. The 78F2, 78P4, and 78C1 are human-engineered to reduce initial errors, improve debugging speed, and aid concept communication.

First, use the 78F2 Flowchartrix™ to lay out your original concept blocks. Then use the 78P4 Print-Out Designers are next. When you finish flowcharting, lay out the printed reports your program will generate. Then when you write code you blaze through the report generation segments right along with the rest of your program.

Unique 70 x 160 matrix accommodates even proportional-spacing word processor fonts. The 160 column width can handle practically any printer format. The 78P4 is big, 14 1/2 x 22 inches, because we've scaled the cell size to human writing comfort, not machine print, giving nearly twice the character-writing area of other printout design sheets.

Special 5-column area records the program line number of the code which creates each printed line. It shows, at a glance, exactly which line of code creates each line of your report, saving hours of needless search time when you must change the report format (and don't you always have to, sooner or later?)

Every sheet of 78CI gives you 2 form uses for the price of one, use 78CI's full 14 1/2 x 22 column grid area to code-regular program steps. Then for interactive or instructional sections, simply keep your characters within the appropriate CRT indicator lines, and you'll automatically know where every character will show on your CRT screen.

Every matrix cell in the 7 x 11 matrix has a specific label to help you track branch points. When you write program documentation, having a separate reference point for each cell makes your program much easier to describe clearly.

With Flowchartrix, you don't need a shape template to draw remarkably regular logic symbols. Guides for the most-used logic symbols are right in each matrix cell, to help you draw most standard flowchart symbols entirely free-hand.

78P4 Print-Out Designers are next. When you finish flowcharting, lay out the printed reports your program will generate. Then when you write code you blaze through the report generation segments right along with the rest of your program.

Works with your CRT display, no matter what brand you own. Equipped for both 13 and 14 column and 24 line x 80 column display formats.

Available in three versions (one for BASIC languages, one for FORTRAN, another for COBOL), the 78C1 is so powerful we include a 7-page instruction manual with every order.

Every tool in the Stirling/Bekdorf system is surface-engineered to take both pen & pencil without blotching. Our tough, extra-heavy, 22# paper is pure enough to use with critical magnetic ink character readers, and gives you crisp, sharp character with pencil or plastic-tipped pen.

Every part of our system uses eye-comfortable soft blue grids. All grid rulings, tints, and division rules are reproduced in a special shade of blue, easy on your eyes even after hours of continuous programming. If you're a professional programmer, you'll particularly appreciate our improvement over the green lines you've been writing on.

A 3-ring binder is one more of our secrets for your success. All your notes, logic, concepts, flowcharts, code, CRT layouts, printout designs, and documentation can be kept together, in order, in one place. When everything you create stays together, debugging and modification is much simpler.

Order your supply of the world's most advanced software development tools, right now, before you hatch even one more bug.

A7 = BA = A! B! A1+! $$

A7 assigns the value 7 to A, and BA= copies the same value to B. This value is then printed three times: A! prints the value of A; B! prints the value of B; and A1+! prints the value of A[1].

Before we pass on to a discussion of the control structures of Mouse, we need one more concept: clause. A clause is a string which contains quoted strings and expressions, which have been defined above; and complete control structures, which are defined below.

Control Structures

A condition in Mouse is written \( E[C] \); E is an expression, and C is a clause. The condition itself is a clause. E is evaluated, and if its value is greater than zero, C is executed. If E is zero or negative, C is skipped. The Pascal statement:

if \( X > 0 \) then \( Y := X \)

becomes in Mouse:

\( X,[YX,=] \)

Since Mouse distinguishes only \( E > 0 \) and \( E \leq 0 \), more complicated expressions must be devised for different conditions. For example, \( X \geq 0 \) is equivalent to \( X+1 > 0 \), and so the Pascal statement:

if \( X > 0 \) then C

becomes in Mouse:

\( X.1+[C] \)

Similarly, \( X = 0 \) is equivalent to \( (1+X > 0) \) and \( (1-X > 0) \), and so Pascal:

if \( X = 0 \) then C

becomes in Mouse:

\( 1X.1+[1X.-[C]] \)

Now, reconsider the definition of clause. A complete condition \( E[C] \) is both a clause itself, and contains a clause, C. Suppose C is itself a condition, \( F[D] \). If we replace C by \( F[D] \) in the clause \( E[C] \), we obtain the clause \( E[F[D]] \). This demonstrates that the nested condition \( 1X.1+[1X.-[C]] \) of the last example is a legitimate Mouse construction.
A loop in Mouse is written:

\[(C_1 \text{ E} \uparrow \text{C}_2)\]

where \(E\) is an expression and \(C_1\) and \(C_2\) are clauses. This loop is a clause; it may be paraphrased as follows:

**START:**
\[C_1\]

if \(E \leq 0\) goto **EXIT**
\[C_2\]

goto **START**

**EXIT:**

In other words, the loop is executed for as long as \(E > 0\), and the exit test may appear anywhere within the loop. Either \(C_1\) or \(C_2\) may be omitted. \((E \uparrow C_2)\) is equivalent to Pascal:

```pascal
while E > 0 do C2
```

and \((C_1 \uparrow E)\) is equivalent to Pascal:

```pascal
repeat C1 until E \leq 0
```

There may be more than one exit test in a loop. For example, \((C_1 \uparrow E \uparrow C_2 \uparrow F \uparrow C_3)\), in which \(E\) and \(F\) are expressions, is allowed.

We now consider some simple Mouse programs which use these control structures. The following program reads a number \(N\), and prints its factorial, \(1 \times 2 \times 3 \times \ldots \times N\):

```mouse
N=? F1= ( N.\uparrow FF.N.* N= NN.1= = ) F!. $$
```

We can shorten the factorial program by using the top of the stack instead of the variable \(F\). The program then becomes:

```mouse
N=? 1 ( N.\uparrow N.\uparrow NN.1= = ) ! $$
```

In this version, \(I\) puts the value one on the stack; \(N.*\) multiplies the value on top of the stack by \(N\); and \(/\) removes the value from the top of the stack and prints it. We can expand this program so that it continues to ask for data until it reads a number which is less than or equal to zero. Note the nested loop:

```mouse
( "ENTER A NUMBER" M=? M.\uparrow
NM.= 1 ( N.\uparrow N.\uparrow NN.1= = )
"FACTORIAL (" M.\! ") = " ! "!"
) $$
```

Here is a dialogue produced by this program:

```
ENTER A NUMBER 3
FACTORIAL (3) = 6
ENTER A NUMBER 7
FACTORIAL (7) = 5040
ENTER A NUMBER 0
```

---

**ANNOUNCING**

New from DEC
LA34 DECwriter IV
$1,199.00

- 110 or 300 baud, RS 232C serial
- ASCII
- Friction feed, paper width to 15”

Teletype 43  $999.00
- RS 232C, 110 or 300 baud
- Upper/lower case full ASCII
- Pin feed, 12” x 8½” paper

New from DIABLO
DIABLO 1640  $2,690.00
Receive-only  $2,331.00
- 150 characters per second
- RS 232C serial interface
- Adjustable forms tractor
- Upper/lower case option $90.00

DIABLO 1650  $2,779.00
Receive-only  $2,419.00
- High-quality daisywheel printing at 45 cps.
- Metal daisywheel printing at 40 cps.

SOROC IQ 120  $795.00
- RS 232C, upper/lower case full ASCII
- Numeric keypad, protected fields
- Cursor keys plus addressable cursor
- Auxiliary extension port

SOROC IQ 140  $1,250.00
- RS 232C and 20mA current loop
- Extensive editing features
- 25th line terminal status display
- 16 function keys (32 with shift)

To Order: Send certified check (personal or company checks require two weeks to clear) including handling* and 6% sales tax if delivered within California.

*Handling: Less than $2,000, add 2%; over $2,000, add 1%. Everything shipped freight collect in factory cartons with manufacturer's warranty.

Circle 222 on inquiry card. July 1979 © BYTE Publications Inc 203
The program which follows reads numbers into an array and then prints them in reverse order. A is used as an array index, and B is the array. The input is terminated by a negative number.

```
A0=  ( BA.+.?= BA.+1↑ AA.1+=  )
     ( A↑ AA.1-= BA.+! )
     $§
```

We can improve this program so that it will read no more than 25 numbers, the maximum capacity of the array B. Note the use of two exit conditions in the first loop:

```
A0=  ( BA.+.?= BA.+1↑ AA.1+=
      A.25- [ "ARRAY FULL!" ↑ ] )
     ( A↑ AA.1-= BA.+! )
     $§
```

Macroinstructions

A complex algorithm always has a hierarchical structure, and a programming language must provide the means of defining and calling procedures and functions so that programs which have an analogous hierarchical structure can be written. In Mouse, procedures and functions are implemented by macroinstructions. A macroinstruction call is written like this:

```
#M;
```

This is a call to macroinstruction M. There must not be a blank between # and the macro name. In this example there are no actual parameters. When there are actual parameters, they follow the macro name and are preceded by commas. The following is a call to macroinstruction M with parameters X and Y:

```
#M X Y;
```

A macro name is a single letter, so a program may use up to 26 different macroinstructions.

Macro definitions come between the main program and the terminating $$. A macro definition starts with the character $ and the name of the macroinstruction, and is terminated by the character @. There must not be any blanks between $ and the name of the macroinstruction being defined. The definition must be a clause, and definitions cannot be nested. The following program uses a macroinstruction, M, to print a message:

```
#M;
$M "A MESSAGE" @
$$
```

Note that the main program code is terminated by $$, which introduces the first macro definition. As usual, the entire program is terminated by $$. When the interpreter encounters #M; in the program, it substitutes the definition of the macroinstruction, excluding $M or @. Thus when the program above is interpreted, it prints:

```
A MESSAGE
```

Each macroinstruction has its own complete set of local variables, A, B, C, ... Z. Assignments to these variables do not affect the values of main program variables. The program which follows will print the number 3, despite the assignment to N in macroinstruction X:

```
N3= #X ; N !
$X N99= @
$$
```

A macro definition may have up to 26 formal parameters, written %A, %B, %C, ... %Z, with no blank between % and the letter. When a formal parameter is encountered in a definition, the actual parameter is

---

**GenRad needs Trolls! What are Trolls?**

Trolls are a rare breed of engineer, who can work in software as well as hardware. At GenRad we design and build Automatic PC-Board Testers for electronic manufacturing. Our support engineers like to eat a chip data sheet for breakfast, breadboard the chip for lunch and debug its software models at dinner. GenRad trolls are good; as a matter of fact, we think they’re the best!

They help guide our customers through the sometimes bewildering land of testing. If you think that you would enjoy this challenging mix of hardware, software and human contact, set up a meeting with our troll leader by contacting Pat Perillo, GenRad, Inc., 300 Baker Avenue, Concord, MA 01742. Or call Pat (collect) at (617)369-4400, ext. 332.

An equal opportunity employer M/F
substituted as a string. Macroinstruction P in the following program prints the value of its parameter:

```plaintext
#P,3; A5= #P,A.; #P,A;
$P %A! @
$$
```

The call #P,3; is equivalent to 3!, and prints 3. The call #P,A.; is equivalent to A!, and prints the value of A, which is 5. These are known as calls by value, because the macro-instruction can obtain only the value of the parameter. The final call, #P,A;, is equivalent to A!, and prints the address of A. It is known as a call by reference, because the parameter is an address. Calls by reference can be used to return values to the calling program. Macroinstruction A in the following program adds the values of its first two parameters, and returns the result as the third parameter:

```plaintext
#A,2,3,X; #A,3,4,Y; #A,X.,Y.,Z;
$A %C %A %B + = @
$$
```

This program adds 2 and 3, giving X, then adds 3 and 4, giving Y, then adds X and Y, giving Z. Note that the actual parameters in the last call are the values X. and Y. and the address Z.

You must know, when writing a macro call, whether the macro-instruction expects a value or an address. The device mentioned before for exchanging two values can be incorporated into a macro-instruction:

```plaintext
$E %A %B. %B %A. = = @
```

In this case, both actual parameters must be addresses. To use this macro-instruction to interchange the values of X and Y write:

```
#E,X,Y;
```

An array can be passed to a macro-instruction by the address of its first component. The following macro-instruction sums %B components of the array %A, and returns the result as %C:

```plaintext
$S %C0= N0=
 ( %BN..nanoTime() += %C.+= NN.1= )
$$
```

The call $S,%A,3,Z; would store the value of %A+B+C in Z.

A macro-instruction which leaves a value on the stack acts as a function. We can rewrite macro-instruction A, which was defined above, as a function:

```plaintext
$A %A%B+ @
```

The call #A,2,3; is now an operand which leaves the value 5 on the stack. This macro-instruction evaluates the factorial function:

```
$F N%A= 1 ( N.1.N. * NN.1 - ) @
```

The clause #F,5; #F,6; + ! prints 840 (=5!+6!).

A macro definition may include macro calls. Moreover, a macro-instruction may call itself, either directly or indirectly, so that it is possible to define recursive macro-instructions. The factorial function can be defined recursively in this way:

```
$F %A [ %A #F,%A1- ; * ]
1%A-- [ 1 ]
$$
```

The character @ acts like a RETURN statement in BASIC or FORTRAN, and we can use it more than once in a macro definition. Here is a slightly shorter version of the recursive factorial macro that exploits this fact:

```
$F %A [ %A #F,%A1- ; * ] 1 @
```

---

**VAK-1 MOTHERBOARD**

- Designed specifically for use with the AIM-65, SYM-1, and KIM-1 microcomputers
- Standard KIM-4* Bus
- Fully buffered Address and Data Bus
- Provides 8 expansion board slots
- Complete with rigid card-cage
- All IC's are socketed
- Provides separate jacks for one audio-cassette, TTY, and Power
- Completely assembled (except for card-cage)

We manufacture a complete line of high quality expansion boards. Use reader service card to be added to our mailing list, or U.S. residents send $1.00 (international send $3.00 U.S.) for airmail delivery of our complete catalog.

*Product of MOS Technology

**PRICE:** $129.00

We also carry the SYM-1 Microcomputer with manuals $269.00

**PRICE:** $129.00

We also carry the SYM-1 Microcomputer with manuals $269.00

---

2967 W. Fairmount Avenue • Phoenix, AZ 85017 • (602) 265-7564

---

Microcomputer with manuals $269.00

Circle 325 on inquiry card.
This definition may be paraphrased in ALGOL:

\[ F(N) := \]
\[ \text{if } N > 0 \text{ then } N \times F(N-1) \text{ else } 1 \]

If \( %A > 0 \), the conditional clause is executed, and the macroinstruction is terminated by the \( @ \) between the brackets. If \( %A \leq 0 \), the value \( f \) is placed on the stack and the macroinstruction is terminated by the final \( @ \).

If there is no actual parameter corresponding to a formal parameter, the formal parameter is processed as a null string. Actual parameters for which there are no formal parameters are ignored. A call to an undefined macroinstruction is also processed as a null string. These points are illustrated by the following program:

\[
\]

\[ $T \" \%A\%B \">!"@ \$
\]

This program prints:

\[ <> \]
\[ <A> \]
\[ <AB> \]
\[ <AB> \]

In the first call, \( \#T \), there are no actual parameters. In the second call, \( \#T, "A"; \), there is one actual parameter, "A", which is printed. In the third call, both parameters are printed. In the fourth call, there are three actual parameters, of which the third, "C", is ignored because there is no formal parameter %C. The last call is to an undefined macroinstruction U, and has no effect.

Now that all of the control structures of Mouse have been described, we can define clause more precisely. A clause may be: an expression, a literal string, a condition, a loop, a macro call, or a clause followed by another clause. Actual parameters and macro definitions must be clauses. These rules are not precise enough to formally define the syntax of Mouse, but they serve as a guide for the Mouse programmer. Their principal purpose is to ensure that all of the components of a control structure are on the same level. They forbid, for example, the use of \( / \) or \( [ \) without the matching \( ) \) or \( ] \) at the same level in a condition, loop, macro definition or actual parameter.

### Example Programs

We conclude this section of the article with some sample Mouse programs. The first program illustrates mutual recursion. Two macroinstructions are said to be mutually recursive if each one calls the other. The square numbers \( S_n \) (1, 4, 9, 16, ...) and the triangular numbers \( T_n \) (1, 3, 6, 10, ...) can be defined in terms of each other in the following way:

\[
S_n = \begin{cases} 1 & \text{if } n = 0 \\ T_n + T_{n-1} & \text{if } n > 0 \end{cases}
\]

\[
T_n = \frac{(S_{n-1} + 3n - 1)}{2} \text{ for } n > 1
\]

The program, which appears in listing 1, uses mutually recursive macroinstructions \( S \) and \( T \) to compute \( S_n \) and \( T_n \).

The second program, which appears in listing 2, prints prime numbers. It loops indefinitely, calling macroinstruction \( P \) for each integer \( N \) in turn. Macroinstruction \( P \) prints \( N \), if \( N \) is prime. The expression \( \%AF. /F. \%A - I+ \) is equivalent to \( 1 - \%A \) \( \text{mod } F \); it is positive only if \( F \) divides \( %A \) exactly. The program will fail when incrementing \( N \) causes overflow, but the algorithm is so inefficient that there is little danger of this happening.

The third program, shown in listing 3, uses recursive macroinstruction \( V \) to print the words of a song. The British and American versions of this song are different, and I am not sure that the words printed by the program are a correct statement of either version. The limit of nine verses is a restriction imposed by the interpreter described in this article, not an inherent limitation of Mouse.

```
"SQUARE AND TRIANGULAR NUMBERS"
( "ENTER A NUMBER" N? = N. !
  "S(" N. ! ") = " #S,N.; !
  "T(" N. ! ") = " #T,N.; [ "|" ]

$S %A1 = [ #T, %A; #T, %A1 -; + @ ] 1 @
$T %A1 = [ #S, %A1 -; 3 %A1 *1 - + 2/ ] 1 @
$
```

Listing 1: This Mouse program reads an integer and prints the corresponding square number \( S_n \) and triangular number \( T_n \). It uses two mutually recursive macroinstructions \( S \) and \( T \).

```
"PRIME NUMBERS!" N1 = ( NN.1 = = #P,N. ; )
$P F1 = N1 =
  ( FF. 1 = = %AF. -1 %AF./F. * %A -1 + ( NO = 01 ) )
  N. [ %AI "\|" ] @
$
```

Listing 2: This program prints a list of prime numbers. Macroinstruction \( P \) determines whether its parameter \( N \) is a prime number and, if it is, prints it.
The experienced 2114 4K static RAM

From EMM — the industry's largest supplier of 4K static RAMS — a 2114 with a year and a half of delivery behind it. Not a new part. Just a new pin-out of a proven part. 1K x 4 organization. 5V only. Standard 18-pin DIP. It draws only 300 mw, has all the speed you need for microprocessor applications.

EMM SEMI, INC.
A division of Electronic Memories & Magnetics Corporation
3883 North 28th Avenue, Phoenix, Arizona 85017 (602) 263-6202

-most items in stock for immediate delivery. Factory-fresh, sealed cartons.

DATA DISCOUNT CENTER
P.O. Box 100
135-53 Northern Blvd., Flushing, New York 11354, 212/665-6609
N.Y.S. residents add appropriate Sales Tax. Shipping FOB N.Y.
BankAmericard, Master Charge add 3%. COD orders require 25% deposit.
The program which appears in listing 4 prints the moves required to solve the Towers of Hanoi problem for a given number of disks. It uses recursion, and demonstrates the use of strings as actual parameters.

```
"HOW MANY VERSES?"
S? = S.1 10s. - [ #s,v,.s. 1 + ; ]
$V %A1 - [ #n,%a; %b; #w; "!"
   #n,%a; %b; "!" #f; "!"
   #n,%a1 -; %b; #w; "!"
   #v,%a1 -; ] @
$B "GREEN BOTTLES" @
$W "STANDING ON THE WALL" @
$F "IF ONE OF THOSE BOTTLES SHOULD HAPPEN TO FALL"
$N %A9 - [ "NINE" @ ]
%A8 - [ "EIGHT" @ ]
%A7 - [ "SEVEN" @ ]
%A6 - [ "SIX" @ ]
%A5 - [ "FIVE" @ ]
%A4 - [ "FOUR" @ ]
%A3 - [ "THREE" @ ]
%A2 - [ "TWO" @ ]
%A1 - [ "ONE" @ ]
%A [ [ "NO" @ ] @
```

Listing 3: Macroinstruction V calls macroinstructions B, W, F, and N to print one verse of a song, and then calls itself recursively to print the next verse. The number of verses is limited to nine, but the interpreter is quite easily modified to enable the program to print more verses.

```
"TOWERS OF HANOK HOW MANY DISKS?"? D? =
D. [ #h,d, [ "LEFT" , "RIGHT", "CENTER" ]; ]
$H %A1 - [ #h,%a1 - %b,%d,%c;
   "MOVE" %b "TO" %c !"
   #h,%a1 - %d,%c,%b; ] @
```

Listing 4: The famous Towers of Hanoi problem. The program prints a list of the moves required to solve the problem for a specified number of rings.

```
"BUBBLE SORT!" #p,a;
$P "HOW MANY NUMBERS?"? N? = N. 26 - [ "TOO MANY!" @ ]
   "ENTER " N. ! " NUMBERS!" NN. 1 - =
   "INPUT ARRAY!" #f,m,0,n , %a.m. + ! ;
   "SORTED ARRAY!" #f,m,0,n , %a.m. + . ! " !"
$B #f,i, %b1 + , %c,
   %f,j, %c.i ,
   %a,j.1 - + , %a.j. + -
   [ %e, %a.j. + , %a.j.1 + - ]; ]; @
$F %A %B = %C %A., - 1 +
   [ [ %C,%a, - 1 + %d %a.a.1 + = ] @ ]
$E %A %B %A, = = @
```

Listing 5: This is the most elaborate Mouse program presented in this article. The main program contains one macro call (#p,a;) only, and so all 26 variables at the lowest level can be used to store an array. The program sorts the array using the bubble sort algorithm, and then prints it.

The last program, shown in listing 5, is the most elaborate. It reads an array, prints it, sorts it into ascending sequence, and prints the sorted array. The main program prints the title and then calls macroinstruction P to do everything else. The address of A is given to P, and since no variables are used in the main program, all 26 components of array A can be used to store the input data. Thus, the program can sort up to 26 numbers. The program makes extensive use of macroinstruction F, which simulates a for statement. The effect of a call to F is:

```
for %A := %B to %C do %D
```

If %B > %C then the index variable %A will be decremented rather than incremented. Macroinstruction E is the exchange macroinstruction introduced earlier. Macroinstruction B does the actual sorting in two nested loops, using the bubble sort algorithm.

Implementation

The Mouse interpreter is presented in listing 6 in the form of a Pascal program. It is not intended, however, that Mouse should be implemented by that program. Listing 6 is intended to be a machine independent guide to the implementation of Mouse in assembly or machine language. Accordingly, the explanation which follows contains hints as to how the Pascal statements can be translated into machine language. Numbers in parentheses refer to line numbers in listing 6. Some explanation of the meaning of the Pascal statements is given here. (If you need more, consult either Jensen and Wirth (reference 3) or my book (reference 4)).

The program starts with a heading (1). This line merely states that the program is called Mouse and that it uses two files, INPUT and OUTPUT. Lines 12 thru 18 declare the global variables of the program. Global variables can be used anywhere in a Pascal program, even in subroutines. The first five declarations (12 thru 16) define arrays. PROG is an array of 500 characters used to store the text of the Mouse program. The components of PROG are written PROG[1], PROG[2]...PROG[500]. DEFINITIONS is an array of 26 integers. Each nonzero component of DEFINITIONS forms an index for the start of a macro definition in the array PROG. The array CALSTACK is the stack used for calculations. The array STACK is the main stack, used to store the status of the program during the expansion of a macro call, a formal parameter, or a loop. Each component of STACK is a variable of type...
FRAMETYPE. This type is defined in lines 5 thru 9. A FRAMETYPE variable has three components: TAG, POS, and OFF. A TAG has one of three values: MACRO, PARAM, or LOOP; these values can be coded as 0, 1, 2, or -1, 0, +1. POS and OFF are both integers. The most convenient way to represent STACK is as a block of 20 units, each unit having three words. Within a unit, the word required is addressed by its offset: 0 for TAG, 1 for POS, and 2 for OFF. For example, the address STACK[N].OFF (the address of component OFF of the Nth unit) is STACK+3*(N-1)+2. Finally, the array DATA is used to store the values of both local and global variables of the Mouse program. The sizes of the arrays are adequate for simple Mouse programs, including all programs in this article. The choice of array sizes is discussed below in greater detail.

Lines 17 and 18 declare global scalar variables. CAL is a pointer to CALSTACK; it is an index for the next free stack word. LEVEL performs the same function for STACK, the main stack. CHPOS is a pointer to the array PROG; it indexes the character currently being processed. The current character itself is stored in CH. Assuming that a single byte is used to store a character, and two bytes are used to store an integer, global data for the interpreter occupies 1248 bytes of memory.

Lines 20 thru 84 define the subroutines used by the interpreter. The function NUM (20 thru 23) maps letters into integers. The Pascal function ORD means simply ordinal value of. The ordinal value of a character is its ASCII (or other) code. For example, if the code for A is 65, then ORD('A') = 65. Thus, NUM('A') = 1, NUM('B') = 2, and so on. The interpreter assumes that the letters have consecutive codes, and hence that NUM('Z') = 26. The function VAL performs a similar task for digits. VAL('0') = 0, VAL('1') = 1, and so on. In machine language, these functions can be implemented by a single instruction which subtracts the appropriate constant from the character value. The procedure GETCHAR (30 thru 33) increments the character pointer CHPOS and sets CH to the next character in the array PROG. The Mouse program is accessed by means of this procedure only.

PUSHCAL (35 thru 38) and POPCAL (40 thru 43) are used to store and remove values from the calculation stack CALSTACK. The parameter of PUSHCAL is the value which is to be pushed onto the stack. The value returned by POPCAL is the value removed from the stack. In a machine language implementation, these values can be passed in a register. PUSH (45 thru 51) and POP (53...
Listing 6 continued from page 209:

```
until (THIS = '$') and (LAST = '$')
end;

begin
LOAD;
CHPOS := 0; LEVEL := 0; OFFSET := 0; CAL := 0;
repeat
  GETCHAR;
  case CH of
    ' ', '!', '0':
      begin
        TEMP := 0;
        while CH in '0'..'9' do
          begin
            TEMP := 10 * TEMP + VAL(CH);
            GETCHAR
          end;
        PUSHCAL(TEMP); CHPOS := CHPOS - 1
      end;
      begin
        TEMP := NUM(CH) + OFFSET;
        PUSHCAL(TEMP); CHPOS := CHPOS - 1
      end;
    '?' :
      begin
        read(TEMP); PUSHCAL(TEMP)
      end;
    '!' write(POPCAL : 1);
    '+' PUSHCAL(POPCAL + POPCAL);
    '-' PUSHCAL(POPCAL + POPCAL);
    '*' PUSHCAL(POPCAL * POPCAL);
    '/' begin
      TEMP := POPCAL;
      PUSHCAL(POPCAL div TEMP)
    end;
    '=' begin
      TEMP := POPCAL;
      DATA[POPCAL] := TEMP
    end;
    '*' begin
      GETCHAR;
      if CH = '!' then writeln
        else if CH = '"' then write(CH)
      repeat
        GETCHAR;
      until CH = '"';
    end;
  end;
end;

('[' if POPCAL <= 0 then SKIP(['',']'));
('[' PUSHLOOP);
('[' if POPCAL <= 0 then
  begin
    POP; SKIP('[',''];
  end;
('[' begin
  CHPOS := STACK(LEVEL).POS;
('#' begin
  GETCHAR;
  if DEFINITIONS(INUM(CH)) > 0 then
    begin
      PUSH(MACRO);
      CHPOS := DEFINITIONS(INUM(CH));
      OFFSET := OFFSET + 26
    end;
  else SKIP('#',');
end;
```

On entry, SKIP assumes that the first character, [, in the example, has been read, so CNT is initially set to 1. Subsequently, [ increments CNT and ] decrements CNT. The procedure terminates when CNT = 0. This procedure is used for skipping over sequences such as:

```
[ ... [ ... ] ... ]
```

The procedure LOAD (73 thru 84) has two functions: it reads the program from the input file into the array PROG, one character at a time; and it stores pointers to macro definitions in the array DEFINITIONS. It uses local variables THIS, LAST, and CHARNUM (74), and it initializes all components of the array DEFINITIONS to zero (76), so that the interpreter can later recognize a call to an undefined macro. LOAD recognizes a macro definition by the sequence $<letter>$ and the end of the program by the sequence $$. In this version of the interpreter, both the Mouse program and its input data are read from the system input file. If you have auxiliary storage, such as disks or cassette tapes, it is probably better to store the Mouse program there. If you use disks or tapes, READ(THIS) on line 79 will call a procedure that reads one character from the chosen device. It is much easier to develop a Mouse program if it is stored on an external medium where it can be attacked with a text editor, rather than keying it directly into memory. Alternately, if you have room in memory, you can elaborate the procedure LOAD into a Mouse editor, rather than using it simply as a loader.

The main program begins at line 86. Initialization consists of loading the program (87) and setting various global variables to zero (88). The rest of the interpreter consists of a single repeat statement (89 thru 197). The body of the repeat statement contains two statements: GETCHAR (90),
Computer Lab of New Jersey

Computer Lab sells the best S-100 Bus products at the best possible prices. Not only are our prices great, so is our delivery. We offer a 10% discount on most major lines, plus a 5% additional discount for a cash purchase.

<table>
<thead>
<tr>
<th>LIST PRICE</th>
<th>OUR CASH PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imsai S10 2-2 Kit</td>
<td>$156.00</td>
</tr>
<tr>
<td>SSM VB-2 Video Board Kit</td>
<td>$149.95</td>
</tr>
<tr>
<td>Vector Graphic Bitstreamer I/O Board</td>
<td>$195.00</td>
</tr>
<tr>
<td>Problem Solver TLC Intelligent CRT Terminal</td>
<td>$1095.00</td>
</tr>
<tr>
<td>Cromemco Z-80 CPU Board Kit</td>
<td>$295.00</td>
</tr>
<tr>
<td>S.D. Sales Z-80 Single Card Computer Kit</td>
<td>$239.00</td>
</tr>
</tbody>
</table>

Subject to available quantities. Prices quoted include cash discount. Shipping and insurance extra.

Call for our prices on:
- Cromemco
- Godbout
- Imsai
- Problem Solver
- SSM
- Vector Graphic
- Tarbell
- Thinker Toya
- Soroc
- California Computer Systems
- Seattle Computer Products
- and many other fine products.

Computer Lab of New Jersey
141 Route 46 • Budd Lake, N.J. 07828
Phone: (201) 691-1984
HOURS: Monday & Friday: 12 to 6, Tuesday-Thursday: 12 to 9
Saturday: 10 to 5

Call or write for our free catalog & price list.

---

16K STATIC RAM

$310
Assembled, Tested, Guaranteed to operate at 4M Hz clock for one year.

Also available with 250 nsec Memory Chips add $35

- Static TMS 4044 or equivalent
- Fully Static 4Kx1 Memory Chips for full DMA capability, no tricky timing problems.
- Fully S-100 Bus Compatible
- All lines fully buffered. Dip Switch Addressable in two 8k block, 4K increments. Write Protectable in 2 blocks, Memory Disable using Phantom. Battery back up capability.
- Bank Select - Using output port 4OH (Cromemco software compatible-addressable to 512KB of Ram for time share or Memory Overlap, also has alternate port 80H.
- Guaranteed - Parts and labor for one year. You may return the undamaged board within 10 days for a full refund.
- Orders - You may phone for Visa, MC. COD ($4 handling charges for COD) orders. Personal checks must clear prior to shipping. Shipping-Stock to 72 hours normally. Will notify expected shipping date for delay beyond this. Illinois residents add 5% tax. Please include phone number with order.

S.C. Digital
P.O. Box 906
Aurora, IL 60507
Phone: (312) 897-7749

---

RADIO SHACK COMPUTER OWNERS

TRS-80™ MONTHLY NEWSLETTER

- PRACTICAL APPLICATIONS
- BUSINESS
- GAMBLING • GAMES
- EDUCATION
- PERSONAL FINANCE
- BEGINNER'S CORNER
- NEW PRODUCTS
- SOFTWARE EXCHANGE
- MARKET PLACE
- QUESTIONS AND ANSWERS
- PROGRAM PRINTOUTS
- AND MORE

FREE

WORD PROCESSING PROGRAM
(Cassette or Disk)
For Writing Letters, Text, Mailing Lists, Files, Etc.
With Each New Subscriptions or Renewal

FREE

24 HOUR ORDER LINE
(914) 425-1535

---

FREE SOFTWARE CATALOGUE (Including listings of hundreds of TRS-80 programs available on cassette and diskette.)

SIGNATURE ____________________________________________ EXP. DATE ______

NAME ____________________________________________

ADDRESS ____________________________________________

Send for FREE Software Catalogue (Including listings of hundreds of TRS-80 programs available on cassette and diskette.)
and a case statement (91 thru 196) which selects an action according to the character obtained by GETCHAR. The repeat statement, and hence the interpreter, terminates when \( \text{CH} = \text{'}\$' \). The operation of the interpreter can therefore be described solely in terms of the action taken for each character returned by GETCHAR. The case statement can be implemented by comparing \( \text{CH} \) to each legal character in turn. A jump table addressed by the ordinal value of \( \text{CH} \) is more efficient, but it may use more memory.

The characters (\( \text{blank} \), \( \text{'}\) , and \( \text{'}\$' \) require no action by the interpreter (93). If the character is \( \text{'}\$' \), the repeat statement terminates, otherwise control returns to GETCHAR (90) which gets the next character.

If the character is a digit (95), this digit and succeeding digits are read, and the value of the corresponding number is accumulated in \( \text{TEMP} \) (97 thru 101). The value of \( \text{TEMP} \) is then pushed onto the stack (102). The interpreter has now read one character past the last digit, which it has to do in order to recognize the end of the number, so the character pointer is backspaced (102). If the character is a letter (105 and 106), its address \( \text{NUM} (\text{CH}) + \text{OFFSET} \) is pushed onto the stack. If \( \text{OFFSET} = 0 \), the address of A is 1, the address of B is 2, and so on. (The use of \( \text{OFFSET} \) will be explained later.) The remaining operand is \( \? \) (109 thru 111), which reads a number from the input file and pushes its value onto the stack. The Pascal statement \( \text{READ(TEMP)} \) reads a signed number from the input file, having skipped over leading blanks.

Lines 113 thru 131 define the actions taken when an operator has been read. Operators use the calculation stack \( \text{CALSTACK} \). The character \( \! \) (113) pops the top value off the stack and prints it. The Pascal statement \( \text{WRITE(POPCAL)} \) prints the value of \( \text{POPCAL} \) without leading or trailing blanks. If blanks are required to separate numbers, they must be explicitly coded in the Mouse program.

The arithmetic operators (115 thru 124) remove two operands from the stack, apply the appropriate operation, and push the result onto the stack. Note that the second operand is on top of the stack, but must be used after the first operand in the noncommutative subtraction and division operations. The Pascal operator \( \div \) means “divide and truncate” (123). The operator \( . \) (period) replaces the address on the stack by the corresponding component of the array \( \text{DATA} \) (126). The assignment operator \( (=) \) uses the address and value on the stack to update a component of the array \( \text{DATA} \) (128 thru 131).

All of the operations on \( \text{CALSTACK} \) are
A landmark book by John L. Stewart, with technical information never before available. Based on years of research with partial support through the original Air Force Bionics program of the 1960’s.

Now you can understand and build devices that behave like the human ear—or a thing called “Hal”.

- Speech recognition for the robot
- Aids for the deaf and deaf-blind
- Reliable voice control
- Speech bandwidth compression
- Speech security
- Binaural localization
- Bat-like echoranging
- Sonar target recognition
- Control of animals with sounds
- Language training
- Speech enhancements
- and much, much more . . .

Hardware representations give new understanding of man—not only how, but also why the ear works as it does.

A far-reaching new theory—complete—sophisticated—powerful—destined to influence not only computers, but also concepts in physiology for decades to come.

$2695* ready to run

PASCAL MICROENGINE™

$2695* ready to run

PASCAL MICROENGINE™

* LIMITED TIME cash price
10% DOWN; guarantee on priority

MICROENGINE™ specifications:
- 16-bit P-code CPU
- 64K bytes RAM
- Floppy disk controller
- Full DMA
- Floating-point hardware
- 2 parallel I/O ports
- 2 parallel I/O ports
- Pascal and Basic compilers
- Text editor
- File manager
- CPU and memory diagnostics
- Symbolic Pascal debugger

Also available:
- Complete turn-key systems
- CRT terminals
- High-speed printers
- Text-quality printers
- Floppy disk drives
- Applications software

Customer satisfaction guaranteed. Full refund with the return of any product within 10 days. Complete technical catalog $1.

5710 Drexel Avenue
Chicago, IL 60637
(312) 684-3183

COVOX Publishing
P.O. Box 2342
Orcutt, CA 93454
Tel.: (805) 937-9545

Expand your TRS-80.
Save $100.

Meet the Vista V80 Mini Disk System. The perfect way to upgrade your TRS-80* system. Inexpensively. (Our $395.00 price is about $100.00 less than the manufacturer's equivalent.) Here's how it can help you.

23% more storage capacity. Useable storage capacity is increased from 55,000 to 65,000 bytes on drive one.

8 times faster. While electronically equal to the TRS-80 Mini-Disk system, track-to-track access is 5ms versus 40ms for the TRS-80.

Better warranty. The V80 carries a 120 day warranty—longer than any comparable unit warranty available.

The Vista V80 Mini Disk System comes complete with Minifloppy disk drive, power supply, regulator board and case. And it's ready to run—simply take it out of the box, plug it in and you're ready to go.

Vista
1320 East St. Andrews Place, Suite I
Santa Ana, California 92705
(714) 558-8813

*TRS-80 ©Tandy Corp.
coded using the subroutines PUSHCAL and POPCAL. They can be coded more efficiently without these subroutines. For example, the action required for division can be written:

\[
\begin{align*}
\text{CAL} & := \text{CAL} - 1; \\
\text{CALSTACK}[\text{CAL}] & := \text{CALSTACK}[\text{CAL}] \\
& \text{div} \text{CALSTACK}[\text{CAL}+1]
\end{align*}
\]

The advantage of using the subroutines PUSHCAL and POPCAL is that they can check for stack underflow (\( \text{CAL} \leq 0 \) in POPCAL) and stack overflow (\( \text{CAL} \geq 20 \) in PUSHCAL), although these checks are not shown in this listing.

When the double quote character " is encountered, the interpreter prints successive characters up to, but not including, the next quote character (133 thru 139). The Pascal procedure WRITELN writes a carriage return/line feed to the output file, and WRITE(CH) writes the single character CH.

The left bracket [ introduces a conditional clause. The value on top of the stack is removed and examined. If it is positive and nonzero, no action is taken, and the interpreter proceeds to execute the bracketed clause. If the value on the stack is zero or negative, the interpreter skips to the matching right bracket ]. The use of the procedure SKIP enables the interpreter to process nested conditions correctly.

Loops are implemented by lines 143 thru 151. The effect of the left parenthesis ( is simply to push a stack frame of type LOOP onto the main stack. This stores the current value of CHPOS on the stack. (It also stores the current value of OFFSET, but OFFSET is not used for loops.) When the interpreter encounters the up arrow symbol (↑), it removes and examines the value on top of the calculation stack. If this value is positive and nonzero, there is nothing to do, but if it is zero or negative, the interpreter must exit from the loop. It does this in two steps (148). First, the main stack is popped. This restores the value of CHPOS, which now points to the left parenthesis ( at the beginning of the loop. Then the procedure SKIP is used to skip over the body of the loop and leave CHPOS pointing to the closing right parenthesis ). This is a slightly inefficient method of terminating the loop, because the entire body of the loop is skipped, rather than just the section from ↑ to ). When the right parenthesis is encountered during the execution of the loop, CHPOS is set to the stacked value POS, which causes the interpreter to jump back to the opening parenthesis. The stack is used for loops to allow loops to be nested.

The rest of the case statement, lines 153 thru 194, handles macro expansion and parameter substitution. When the interpreter encounters the character # (153), it reads the character which follows. This character should be a letter (154). If there is a definition for the macroinstruction (DEFINITIONS[NUM(CH)] > 0), the interpreter pushes a MACRO entry onto the main stack (158) and assigns new values to CHPOS (159) and OFFSET (160). CHPOS now points to the first character of the macro definition. The effect of adding 26 to OFFSET is to allocate 26 local variables in the array DATA for the use of the macro. The address of the local variable A is NUM('A')+OFFSET (see line 107); in the main program this is 1, in a macro-instruction called from the main program it is 27, and so on. If there is no definition for the macroinstruction, the interpreter skips to the semicolon which terminates the call (162). The procedure SKIP must be used to find the semicolon because the actual parameters of the macro call may include macro calls. The interpreter continues to process the macro definition until it encounters an @, at which point it pops the main stack (166). Popping the stack resets OFFSET.
25 START-AT-HOME COMPUTER BUSINESSES

In "Low Capital, Startup Computer Businesses"

CONSULTING • PROGRAMMING • MICRO COMPUTER OPPORTUNITIES • SOFTWARE PACKAGES • FREELANCE WRITING • SEMINARS • TAPE/DISC CLEANING • FIELD SERVICE • SYSTEMS HOUSES • LEASING • SUPPLIES • PUBLISHING • HARDWARE DISTRIBUTORS • SALES AGENCIES • USED COMPUTERS • FINDER'S FEES • SCRAP COMPONENTS • AND MORE...

Plus — ideas on moonlighting, going full-time, image building, revenue building, bidding, contracts, marketing, professionalism, and more. No career tool like it. Order now — if not completely satisfied, return within 30 days for full immediate refund.

* 6½ x 11 ringbound * 156 pp. * $20.00
Phone Orders 901-761-9090

DATASEARCH
incorporated

4954 William Arnold Road, Dept. B, Memphis, TN 38117
Rush my copy of "Low Capital Startup Computer Businesses" at $20.

NAME/COMPANY
ADDRESS
CITY/STATE/ZIP

□ Check Enclosed □ VISA □ Master Charge

Phone Orders 901-761-9090

EASYSOFTWARE

for your APPLE
Let your APPLE teach you
FRENCH, GERMAN, ITALIAN, and SPANISH

Each package contains more than 1000 French, German, Italian or Spanish words and their English equivalents. You may answer in French, German, etc. or English either by multiple choice or by fill-in. The programs keep track of correct and incorrect responses and sort words according to their difficulty.

Then enjoy COMPUTER ART new beautiful symmetrical high resolution color designs. An almost infinite number! Endless hours of enjoyment. Don't get high and start watching these!

( ) FRENCH $25 Name
( ) GERMAN $25 Address
( ) ITALIAN $25 City
( ) SPANISH $25 State
( ) HI ART $10 Zip

Send check or money order to:
EASYSOFTWARE
PO Box 84305
Veterans Adm Br
Los Angeles CA 90073

MICROPOLIS SOFTWARE (MOD II)

WHY
Do you need the CCA DATA MANAGEMENT SYSTEM?
BECAUSE
With this powerful system,
You’re in control

- YOU decide on the files you need (Name & Address A/R, A/P, Payroll, Inventory, Customer Lists, etc.)
- YOU choose the names for the files and all of the data fields in the files.
- YOU let the DMS assist you in maintaining your files using YOUR field names. Records can be added, updated, deleted, inspected, or scanned for in any file.
- YOU let the DMS sort your files as needed.
- YOU select the report or mailing label format, selecting title, fields, totaling, sub-totaling, editing and more.

Stay in control!

Order the CCA Data Management System today!
Only $150.00 - Comprehensive Documentation

Creative Computer Applications
2218 Glen Canyon Road
Altadena, CA 91001
(213) 798-4529
correctly, but it leaves CHPOS pointing at the macroinstruction name. The call to SKIP (166) moves CHPOS past the macro call so that processing can continue.

The most difficult task for the interpreter is processing a formal parameter (169 thru 192). First, the interpreter reads the parameter name and sets PARNUM to the corresponding numeric value (170). For example, if it reads %A, PARNUM is set to 1. A new entry is created in the main stack. The next problem is to find, in the stack, the stack frame created by the corresponding macro call. This would be easy if the call frame was at the top of the stack, but this is not necessarily the case.

Consider the situation shown in figure 1, which shows a simple Mouse program and the main stack during its execution. The interpreter reads %A, creates the stack frame at level 1, and moves CHPOS to $A. It then reads %B, creates the stack frame at level 2, and moves CHPOS to $B. Now it encounters %A, creates a PARAM stack frame at level 3, and sets CHPOS to the actual parameter. The actual parameter is %A (in the call %B,%A;) and so it creates a new stack frame at level 4. Now the interpreter has to find the actual parameter corresponding to this %A.

In order to do this it must locate the stack frame at level 1, which contains a pointer to %A. It can find the correct frame by using the fact that MACRO and PARAM frames in the stack are nested. (The situation is slightly more complicated when an actual parameter contains a macro call, but the same strategy works.) The search is implemented by lines 171 thru 179 of the interpreter. LOOP frames on the stack are ignored (177).

When the interpreter has found the correct stack frame, it sets the values of CHPOS and OFFSET from it (180 and 181). The stacked value of OFFSET must be used because variables in an actual parameter belong to the level of the macro call, not to the level of its definition. CHPOS now points to the name of the macro in the macro call. The interpreter finds the correct parameter by counting PARNUM commas (182 thru 190).

The counting process is complicated by two factors. One is that the actual parameter may contain macro calls; this contingency is handled by SKIP (184 thru 188). The other complication is that there may be no actual parameter corresponding to the formal parameter. In this case, the interpreter will encounter a semicolon (190), and must pop the stack frame that it just created (191).

An actual parameter is terminated by either a comma or a semicolon. The action of the interpreter is simply to pop the main stack (194).

Improving the Implementation

The Mouse interpreter that is presented here has been pared to the bare essentials. It is complete and accurate, and was used to test the example programs of listings 1 thru 5. Mouse is easier to use, however, if the interpreter does some error checking.
Thanks to these inexpensive Pet and TRS-80 memories and disk drives you can instantly turn your personal computer into a speedy and efficient professional computer.

**Expand your Pet.**
1. **Disk Drive.** Double density 400k on-line disk drive and DOS operating system including 17 additional BASIC commands. Random or indexed sequential access. Exceptionally fast — loads 20k program in 3 seconds. Only $1295.
2. **Expandamem™** Completely internal memory that adds up to 32k. 32k unit allows for 8k of assembly language subroutines accessed via the USR command. 16k only $425. 24k only $525. 32k only $615.
3. **Software.** Complete library available. Write for details.

**Expand your TRS-80.**
1. **Disk Drive.** Compact dual disk drive is completely compatible with TRS-80 DOS and uses own controller. Only $895.

---

**IMMEDIATE DELIVERY! ORDER TODAY!**

**COMPU/THINK™**
3260 Alpine Road, Menlo Park, CA 94025
(415) 854-2577

Please send me the following:
- Pet Disk Drive
- Pet Expandamem™ 16k 24k 32k
- Pet Software (Info only) Disk Drive

Total Enclosed: $
Check or Money Order only. Calif. Res. add 6% tax. Freight pre-paid in continental U.S.

Name __________________________
Address _________________________
City __________________ State ____ Zip ______

Dealer Inquires Invited.
Some of the errors which it can easily detect are listed here:

- Stack underflow and overflow (both stacks). Underflow is always the result of an error in the Mouse program; overflow may be due to an error in the Mouse program, but it is more likely that the stack is not large enough for the program. Checks for underflow should be incorporated into \texttt{POP} and \texttt{POPCAL}, and checks for overflow should be incorporated into \texttt{PUSH} and \texttt{PUSHCAL}.
- Illegal characters. The interpreter should check the program for illegal characters; this can be done during loading.
- Division by zero. \texttt{TEMP} = 0 at line 122.
- Illegal address. The value of the stack at line 126 should satisfy 1 \leq \texttt{POPCAL} \leq \texttt{OFFSET}+26; the same test can be made at line 129. A stricter address check would be preferable, but is not easy to devise, since a macroinstruction can access the variables both at its own level and at lower levels by means of parameters.
- Context errors. The characters \# and \% must always be followed by a letter.
- Undefined macroinstruction or missing actual parameter. These would be regarded as errors in a stricter implementation of Mouse.

A tracing option is a powerful aid to debugging Mouse programs. The easiest way to trace the execution of a Mouse program is to make the interpreter display the value of each character it processes. This can be done by inserting \texttt{WRITE(CH)} after \texttt{GETCHAR} at line 90 of the interpreter (listing 6). It is also useful to trace the results of assignments. This can be done by printing the value of \texttt{TEMP} at line 129.

The size of the arrays (12 thru 16) can be adjusted to suit your requirements. Most expressions can be evaluated with a small stack of two or three entries, and you may find it surprising that \texttt{CALSTACK} has space for so many entries. The reason is that some recursive macroinstructions (such as \texttt{S} and \texttt{T} in listing 1) create an entry in the calculation stack at each level of recursion, and \texttt{CALSTACK} must be large enough to hold these. The array \texttt{DATA} makes the poorest use of space; 26 words are allocated at each level of macrocall. This implementation allows ten calling levels, which is not as generous as it sounds if you are using recursive macroinstructions. You can reduce the space requirement to ten variables (A,B, ... J).
at each level, by changing line 160 to
OFFSET := OFFSET + 10. Note that a
macroinstruction with no local variables
needs space in the array STACK but not in
the array DATA, so recursive macros such as
S and T in listing 1, V in listing 3, and H in
listing 4, are limited only by the size of the
arrays STACK and CALSTACK.

Improving the Language

It is easy to add features to Mouse. A
random number generator is useful, particu­
larly for programming games. Probably the
simplest method is to use a unary operator
which multiplies the number on the top of
the calculation stack by a real random num­
ber R such that 0 < R < 1, truncates the
result, and increments it. If the character :
(colon) is used to denote the operator, then
6: would leave a simulated die throw on the
stack.

The most severe restriction of this partic­
ular version of Mouse is that it cannot
process character data. A more powerful
version of Mouse can be obtained by redefin­
ing ? and! so that they read and write a
single character. The disadvantage is that
macroinstructions are then required to read
and print numbers—not a large price to pay
for the greater generality achieved.

It is quite easy to add a case construction
to the language. The following syntax is
suitable:

E< C_1, C_2, ..., C_n >

Each C_i is a clause. When the interpreter
reads <, it performs the following actions:

- skip to the matching > and push a
  CASE frame onto the main stack;
- return to <;
- if E > 1, then scan the clause list until
  the (n-1)th comma is encountered,
  otherwise pop the stack. If the charac­
ter > is encountered during this scan,
  then E > n, and the case clause is
  null or illegal.

The action for comma, which is already
defined to be POP, is correct. This construc­
tion will select and execute one of the
clauses C_1, C_2, ..., C_n, according to the value
of E. It is very easy to write a random sen­
tence generating program in a version of
Mouse to which a random number generator
and a case construction have been added.

The facilities for annotating Mouse pro­
grams are very limited. Strings in quotes may
be used in the main program outside loops.
These serve as comments to the program.

THE
ONLY
6800/09
USER-ORIENTED
MAGAZINE

MAGAZINE COMPARISON

<table>
<thead>
<tr>
<th>(2 years)</th>
<th>Monthly Averages</th>
<th>TOTAL PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>KB BYTE CC DOBB'S</td>
<td>7.8 64 27 22</td>
<td>191 ea. mo</td>
</tr>
<tr>
<td>Average cost for all four each month $6.88</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(based on advertised 1-year subscription price)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$9.60 cost per month, life</td>
<td></td>
<td></td>
</tr>
<tr>
<td>($10.50 Charter Subscription Rate)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>THAT'S RIGHT! MUCH, MUCH MORE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>for 1/6 the Cost!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHARTER SUBSCRIPTION SPECIAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-Year $10.50 2 Years $18.50 3 Years $26.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>OK, PLEASE ENTER MY SUBSCRIPTION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bill My Master Charge or VISA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card Exp. Date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For 1-Year _ 2 Years _ 3 Years</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enclosed $</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name ______________</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Street __________________</td>
<td></td>
<td></td>
</tr>
<tr>
<td>City _______ State Zip</td>
<td></td>
<td></td>
</tr>
<tr>
<td>My Computer is</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

68 MICRO JOURNAL
3018 Hamill Road
HIXSON, TN 37343

FOREIGN ADD: $9.50 Per Yr. Surface $26.50 Per Yr. Air Mail

NOTE: Subscription Rates to increase August 1979 by 40%—Don't miss
anymore than you have already!
text and also as a guide to progress when the program is running, since they are printed by the interpreter. The interpreter ignores text between macro definitions, so macroinstructions can be titled. However, these methods use up valuable space in the array PROG. A better solution is to use a special symbol such as \apostrophe, or a pair of symbols such as \\ and \\, and to modify the loader so that it does not store comments in memory.

Conclusion

Mouse is simple enough to be implemented on a small computer system in a few days, yet it is rich enough to give insight into the mechanisms used by much higher level languages. The Mouse interpreter can be used by itself or as part of a larger system. The General Purpose Macrogenerator is considerably more powerful than Mouse, but nonetheless I think that Strachey's appraisal of the GPM provides an apt conclusion:

It has been our experience that the GPM, while a very powerful tool in the hands of a ruthless programmer, is something of a trap for the unsophisti-

cated one. It contains in itself all the undesirable features of every possible machine code — in the sense of inviting endless tricks and time-wasting though fascinating exercises in ingenuity — without any of the irritating \textit{ad hoc} features of real machines. It can also be almost impenetrably opaque, and even very experienced programmers indeed tend to spend hours simulating its action when one of their macro definitions goes wrong. Furthermore, it is remarkably good at using up machine time — fortunately the programs written for it are usually rather short.

REFERENCES

WE SHIP FROM STOCK—EVERYTHING FACTORY FRESH, FULLY WARRANTED
WE ARE KNOWN FOR OUR PROMPT, COURTEOUS SERVICE
TELETYPE MODEL 43 TTL ........................................... $985
RS232 ................................................................. $1,085
(We stock Teletype Paper and Ribbons)

DEC LA 34
High quality, hard copy, dot matrix printer. Plug compatible substitute for the Teletype 43. RS232 included at no extra cost .......... $1,159
HAZELTINE 1500 assembled ...................................... $945
1510 ..................................................................... $1,085
Also available with French, German or Danish character sets.

MARINCHIP SYSTEMS M9900 CPU
The Complete, Compatible 16 bit CPU for the S-100 Bus
Kit .......................................................................... $550
Assembled ........................................................... $700
We configure systems to suit your individual requirements.

MODEM
Originater/Answerer. The "CAT" from Novation .................. $190
HARD DISK CONTROLLER From Konian ...................... $1,550
S-100 compatible; plugs into S-100 mainframe, controls 1-4 drives.

We Also Carry:
* PER SCI* FUJITSU HARD DISK DRIVES*TEI MAINFRAMES*
* TARBELL CONTROLLERS* INTERTUBE* SOROC*
* IMS AND OTHER MEMORIES* TELL US OF YOUR NEEDS*
* WE CAN OBTAIN MANY ITEMS AT LOW COST*
To Order: $12 shipping for Terminals. $4 shipping for other items. 24 hr.
shipping upon receipt of certified check or money order. Personal checks:
allow 10 days. Credit cards: 4% charge. NY residents add tax.

We EXPORT TO ALL COUNTRIES
OVERSEAS CALLERS USE (212) 448-6298 ONLY
We have no reader inquiry number. Please call or write.

JOHN D. OWENS ASSOCIATES, INC.
147 NORWOOD AVENUE
STATEN ISLAND, NEW YORK 10304
DAY, EVENING, WEEKEND, HOLIDAY CALLS WELCOME!
(212) 448-6283

SOFTWARE
FOR BUSINESS
includes:
* Mailing List
* General Ledger
* Payroll
* Phone Directory
* Customer Information
* Invoice Writer
* Inventory
* Check Writer
* Complete Business System
* Master Business System
Available Soon Word Processor

2224 N. University, Peoria, IL 61604
Phone 309/686-9352

*Trademark of Apple Computer Co. Inc.
# Intel 8080 Microprocessor Instruction Set

In my work with the 8080 microprocessor, I have found the accompanying instruction set summary very useful. The instructions are grouped in five tables according to function, with a single line summary being given for each instruction. In addition, there is an explanatory table of nomenclature and symbols. The order of the instructions is the same as given in the Intel 8080 Microcomputer Systems User's Manual. The method of tabulation makes it very clear which registers and flags are affected by the execution of each instruction.

---

## Symbol Meaning

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8 bit accumulator</td>
</tr>
<tr>
<td>B, C, D, E, H, L</td>
<td>8 bit general purpose registers</td>
</tr>
<tr>
<td>F</td>
<td>Condition code flags Z, S, P, CY, AC (Zero, Sign, Parity, Carry, and Auxiliary Carry)</td>
</tr>
<tr>
<td>PSW</td>
<td>16 bit Processor Status Word (comprising A and F)</td>
</tr>
<tr>
<td>rp</td>
<td>One of the register pairs B representing BC, D representing DE, H representing HL, or SP representing 16 bit stack pointer</td>
</tr>
<tr>
<td>r, r1, r2</td>
<td>One of the registers A, B, C, D, E, H, L, or M</td>
</tr>
<tr>
<td>m</td>
<td>bit m of the register r</td>
</tr>
<tr>
<td>addr, data16</td>
<td>16 bit address or data quantity (note: low byte is stored first)</td>
</tr>
<tr>
<td>pp</td>
<td>8 bit port number</td>
</tr>
<tr>
<td>n</td>
<td>Restart number 0 thru 7</td>
</tr>
<tr>
<td>cc</td>
<td>One of the condition code tests NZ not zero Z = 0, Z zero Z = 1, NC no carry CY = 0, C carry CY = 1, PO parity odd P = 0, PE parity even P = 1, P plus S = 0, M minus S = 1</td>
</tr>
<tr>
<td>( )</td>
<td>Indirect reference</td>
</tr>
<tr>
<td>-(SP)</td>
<td>Stack push operation</td>
</tr>
<tr>
<td>(SP)+</td>
<td>Stack pop operation</td>
</tr>
<tr>
<td>+</td>
<td>is replaced by</td>
</tr>
<tr>
<td>-</td>
<td>is exchanged with</td>
</tr>
<tr>
<td>boolean NOT (bar above symbol)</td>
<td>boolean AND</td>
</tr>
<tr>
<td>boolean OR</td>
<td>boolean Exclusive — OR</td>
</tr>
<tr>
<td>+</td>
<td>Addition</td>
</tr>
<tr>
<td>-</td>
<td>Two's complement subtraction</td>
</tr>
</tbody>
</table>
## Data Transfer Group

<table>
<thead>
<tr>
<th>Flags affected</th>
<th>Registers affected</th>
<th>Bytes</th>
<th>Op code</th>
<th>Operands</th>
<th>Meaning</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z S P CY AC</td>
<td>r2</td>
<td>1</td>
<td>MOV</td>
<td>r2, r1</td>
<td>Move register r1 to register r2</td>
<td>r2 (\rightarrow) r1</td>
</tr>
<tr>
<td></td>
<td>r</td>
<td>2</td>
<td>MVI</td>
<td>r, data</td>
<td>Move immediate data</td>
<td>r (\rightarrow) data</td>
</tr>
<tr>
<td></td>
<td>rp</td>
<td>3</td>
<td>LXI</td>
<td>rp, data16</td>
<td>Load register pair immediate</td>
<td>rp (\rightarrow) data16</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>3</td>
<td>LDA</td>
<td>addr</td>
<td>Load accumulator direct</td>
<td>A (\rightarrow) (addr)</td>
</tr>
<tr>
<td></td>
<td>HL</td>
<td>3</td>
<td>STA</td>
<td>addr</td>
<td>Store accumulator direct</td>
<td>(addr) (\leftarrow) A</td>
</tr>
<tr>
<td></td>
<td>HL, DE</td>
<td>1</td>
<td>XCHG</td>
<td></td>
<td>Exchange HL with DE</td>
<td>HL (\rightarrow) DE</td>
</tr>
</tbody>
</table>

### Arithmetic Group

- Add register: A \(\leftarrow\) A + r
- Add immediate: A \(\leftarrow\) A + data
- Add register with carry: A \(\leftarrow\) A + r + CY
- Add immediate with carry: A \(\leftarrow\) A + data + CY
- Subtract register: A \(\leftarrow\) A - r
- Subtract immediate: A \(\leftarrow\) A - data
- Subtract register with borrow: A \(\leftarrow\) A - r - CY
- Subtract immediate with borrow: A \(\leftarrow\) A - data - CY
- Increment register: r \(\rightarrow\) r + 1
- Decrement register: r \(\leftarrow\) r - 1
- Increment register pair: rp \(\rightarrow\) rp + 1
- Decrement register pair: rp \(\leftarrow\) rp - 1
- Add register pair to HL: HL \(\leftarrow\) HL + rp

*The 8 bit number in A is adjusted to form two 4 bit BCD digits.*

---

## TERMINALS FROM TRANSMET

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>PURCHASE PRICE</th>
<th>12 M.O.S</th>
<th>24 M.O.S</th>
<th>36 M.O.S</th>
</tr>
</thead>
<tbody>
<tr>
<td>LA36 DECwriter II</td>
<td>$1,595</td>
<td>$152</td>
<td>$83</td>
<td>$56</td>
</tr>
<tr>
<td>LA34 DECwriter IV</td>
<td>$1,955</td>
<td>$176</td>
<td>$96</td>
<td>$64</td>
</tr>
<tr>
<td>LA120 DECwriter III, KSR</td>
<td>$2,295</td>
<td>$219</td>
<td>$120</td>
<td>$80</td>
</tr>
<tr>
<td>LS120 DECwriter III, RO</td>
<td>$1,995</td>
<td>$190</td>
<td>$104</td>
<td>$70</td>
</tr>
<tr>
<td>LA180 DECprinter I, RO</td>
<td>$1,995</td>
<td>$190</td>
<td>$104</td>
<td>$70</td>
</tr>
<tr>
<td>VT100 CRT DECScope</td>
<td>$1,695</td>
<td>$162</td>
<td>$88</td>
<td>$59</td>
</tr>
<tr>
<td>VT132 CRT DECScope</td>
<td>$1,895</td>
<td>$181</td>
<td>$97</td>
<td>$66</td>
</tr>
<tr>
<td>T1745 Portable Terminal</td>
<td>$1,875</td>
<td>$179</td>
<td>$98</td>
<td>$66</td>
</tr>
<tr>
<td>T1765 Bubble Memory Term.</td>
<td>$2,795</td>
<td>$267</td>
<td>$145</td>
<td>$98</td>
</tr>
<tr>
<td>T1810 RO Printer</td>
<td>$1,895</td>
<td>$181</td>
<td>$99</td>
<td>$66</td>
</tr>
<tr>
<td>T1820 KSR Printer</td>
<td>$2,395</td>
<td>$229</td>
<td>$125</td>
<td>$84</td>
</tr>
<tr>
<td>ADM3A CRT Terminal</td>
<td>$875</td>
<td>$84</td>
<td>$46</td>
<td>$31</td>
</tr>
<tr>
<td>QUME Letter Quality KSR</td>
<td>$3,195</td>
<td>$306</td>
<td>$166</td>
<td>$112</td>
</tr>
<tr>
<td>QUME Letter Quality RO</td>
<td>$2,795</td>
<td>$268</td>
<td>$145</td>
<td>$98</td>
</tr>
<tr>
<td>HAZELTINE 1410 CRT</td>
<td>$895</td>
<td>$86</td>
<td>$47</td>
<td>$32</td>
</tr>
<tr>
<td>HAZELTINE 1500 CRT</td>
<td>$1,195</td>
<td>$115</td>
<td>$62</td>
<td>$42</td>
</tr>
<tr>
<td>HAZELTINE 1520 CRT</td>
<td>$1,595</td>
<td>$152</td>
<td>$83</td>
<td>$56</td>
</tr>
<tr>
<td>DataProducts 2230</td>
<td>$7,900</td>
<td>$755</td>
<td>$410</td>
<td>$277</td>
</tr>
<tr>
<td>DATAMATE Mini Floppy</td>
<td>$1,750</td>
<td>$167</td>
<td>$91</td>
<td>$61</td>
</tr>
</tbody>
</table>

**ACCESSORIES AND PERIPHERAL EQUIPMENT**
- Acoustic Couplers
- Modems
- Thermal Paper Ribbons
- Interface Modules
- Floppy Disk Units

**DIGITAL TIMING DEVICES**

- **New! 779 Line Printer Timer**
  - Works with TRS-80® and Centronics® 779 Line Printers
  - Turns Motor on and off Automatically

No software or hardware changes needed. Saves motor life and power. Just solder 3 wires and mounts inside printer. Dealers wanted, inquire on company stationery, also O.E.M. and service accounts wanted. $75.00 complete with one year warranty. Make checks payable to:

**Digital Timing Devices**

4306 N.E. 6 Ave.
Ft. Lauderdale, FL 33334 USA
Phone # (305) 561-3757

Division of D.S.S.I.

*These are registered trademarks for Radio Shack & Centronics.*
### Logical Group

<table>
<thead>
<tr>
<th>Flags affected</th>
<th>Registers affected</th>
<th>Bytes</th>
<th>Op code</th>
<th>Operands</th>
<th>Meaning</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z S P CY AC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>✓ ✓ ✓ 0 ✓ ✓</td>
<td>A, F</td>
<td>1</td>
<td>ANA</td>
<td>r</td>
<td>AND register</td>
<td>A ← A ∧ r</td>
</tr>
<tr>
<td>✓ ✓ ✓ 0 ✓ ✓</td>
<td>A, F</td>
<td>2</td>
<td>ANI</td>
<td>data</td>
<td>AND immediate</td>
<td>A ← A ∧ data</td>
</tr>
<tr>
<td>✓ ✓ ✓ 0 ✓ ✓</td>
<td>A, F</td>
<td>1</td>
<td>XRA</td>
<td>r</td>
<td>Exclusive —OR register</td>
<td>A ← A ∨ r</td>
</tr>
<tr>
<td>✓ ✓ ✓ 0 ✓ ✓</td>
<td>A, F</td>
<td>2</td>
<td>XRI</td>
<td>data</td>
<td>Exclusive —OR immediate</td>
<td>A ← A ∨ data</td>
</tr>
<tr>
<td>✓ ✓ ✓ 0 ✓ ✓</td>
<td>A, F</td>
<td>1</td>
<td>ORA</td>
<td>r</td>
<td>OR register</td>
<td>A ← A ∨ r</td>
</tr>
<tr>
<td>✓ ✓ ✓ 0 ✓ ✓</td>
<td>A, F</td>
<td>2</td>
<td>ORI</td>
<td>data</td>
<td>OR immediate</td>
<td>A ← A ∨ data</td>
</tr>
<tr>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
<td>F</td>
<td>1</td>
<td>CMP</td>
<td>r</td>
<td>Compare register</td>
<td>A ← r (Z = 1 if A = r) (CY = 1 if A &lt; r)</td>
</tr>
<tr>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
<td>F</td>
<td>2</td>
<td>CPI</td>
<td>data</td>
<td>Compare immediate</td>
<td>A ← data (Z = 1 if A = data) (CY = 1 if A &lt; data)</td>
</tr>
<tr>
<td>✓ ✓ A, CY</td>
<td>1</td>
<td>RLC</td>
<td></td>
<td></td>
<td>Rotate left</td>
<td>A&lt;sub&gt;n+1&lt;/sub&gt; ← A&lt;sub&gt;n&lt;/sub&gt; (A&lt;sub&gt;n&lt;/sub&gt; = A&lt;sub&gt;n&lt;/sub&gt;) CY ← A&lt;sub&gt;n&lt;/sub&gt;</td>
</tr>
<tr>
<td>✓ ✓ A, CY</td>
<td>1</td>
<td>RRC</td>
<td></td>
<td></td>
<td>Rotate right</td>
<td>A&lt;sub&gt;n+1&lt;/sub&gt; ← A&lt;sub&gt;n&lt;/sub&gt; (A&lt;sub&gt;n&lt;/sub&gt; = A&lt;sub&gt;n&lt;/sub&gt;) CY ← A&lt;sub&gt;n&lt;/sub&gt;</td>
</tr>
<tr>
<td>✓ ✓ A, CY</td>
<td>1</td>
<td>RAL</td>
<td></td>
<td></td>
<td>Rotate left through carry</td>
<td>A&lt;sub&gt;n+1&lt;/sub&gt; ← A&lt;sub&gt;n&lt;/sub&gt; (A&lt;sub&gt;n&lt;/sub&gt; = A&lt;sub&gt;n&lt;/sub&gt;) CY ← A&lt;sub&gt;n&lt;/sub&gt;</td>
</tr>
<tr>
<td>✓ ✓ A, CY</td>
<td>1</td>
<td>RAR</td>
<td></td>
<td></td>
<td>Rotate right through carry</td>
<td>A&lt;sub&gt;n+1&lt;/sub&gt; ← A&lt;sub&gt;n&lt;/sub&gt; (A&lt;sub&gt;n&lt;/sub&gt; = A&lt;sub&gt;n&lt;/sub&gt;) CY ← A&lt;sub&gt;n&lt;/sub&gt;</td>
</tr>
<tr>
<td>✓ ✓ A</td>
<td>1</td>
<td>CMA</td>
<td></td>
<td></td>
<td>Complement accumulator</td>
<td>A ← A^-</td>
</tr>
<tr>
<td>✓ ✓ CY</td>
<td>1</td>
<td>CMC</td>
<td></td>
<td></td>
<td>Complement carry</td>
<td>CY ← CY</td>
</tr>
<tr>
<td>✓ ✓ CY</td>
<td>1</td>
<td>STC</td>
<td></td>
<td></td>
<td>Set carry</td>
<td>CY ← 1</td>
</tr>
</tbody>
</table>

### Branch Group

<table>
<thead>
<tr>
<th>Flags affected</th>
<th>Registers affected</th>
<th>Bytes</th>
<th>Op code</th>
<th>Operands</th>
<th>Meaning</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z S P CY AC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PC</td>
<td>3</td>
<td>JMP</td>
<td>addr</td>
<td></td>
<td>Jump</td>
<td>PC ← addr</td>
</tr>
<tr>
<td>PC</td>
<td>3</td>
<td>Jcc</td>
<td>addr</td>
<td></td>
<td>Conditional jump</td>
<td>PC ← addr (if cc true)</td>
</tr>
<tr>
<td>PC, SP</td>
<td>3</td>
<td>CALL</td>
<td>addr</td>
<td></td>
<td>Call</td>
<td>−(SP) ← PC, PC ← addr</td>
</tr>
<tr>
<td>PC, SP</td>
<td>3</td>
<td>CCC</td>
<td>addr</td>
<td></td>
<td>Conditional call</td>
<td>−(SP) ← PC, PC ← addr (if cc true)</td>
</tr>
<tr>
<td>PC, SP</td>
<td>1</td>
<td>RET</td>
<td></td>
<td></td>
<td>Return</td>
<td>PC ← (SP)+</td>
</tr>
<tr>
<td>PC, SP</td>
<td>1</td>
<td>Rcc</td>
<td></td>
<td></td>
<td>Conditional return</td>
<td>PC ← (SP)+ (if cc true)</td>
</tr>
<tr>
<td>PC, SP</td>
<td>1</td>
<td>RST</td>
<td>n</td>
<td></td>
<td>Restart</td>
<td>−(SP) ← PC, PC ← 8n</td>
</tr>
<tr>
<td>PC</td>
<td>1</td>
<td>PCHL</td>
<td></td>
<td></td>
<td>Jump HL indirect</td>
<td>PC ← HL</td>
</tr>
</tbody>
</table>

### Stack, I/O, and Machine Control Group

<table>
<thead>
<tr>
<th>Flags affected</th>
<th>Registers affected</th>
<th>Bytes</th>
<th>Op code</th>
<th>Operands</th>
<th>Meaning</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z S P CY AC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SP</td>
<td>1</td>
<td>PUSH</td>
<td>rp*</td>
<td>Push register pair</td>
<td>−(SP) ← rp</td>
</tr>
<tr>
<td></td>
<td>SP</td>
<td>1</td>
<td>PUSH</td>
<td>PSW</td>
<td>Push processor status word</td>
<td>−(SP) ← A, F</td>
</tr>
<tr>
<td></td>
<td>SP, rp</td>
<td>1</td>
<td>POP</td>
<td>rp*</td>
<td>Pop register pair</td>
<td>rp ← (SP)+</td>
</tr>
<tr>
<td></td>
<td>SP, A, F</td>
<td>1</td>
<td>POP</td>
<td>PSW</td>
<td>Pop processor status word</td>
<td>A&lt;sub&gt;n&lt;/sub&gt; ← (SP)+</td>
</tr>
<tr>
<td></td>
<td>HL</td>
<td>1</td>
<td>XTHL</td>
<td></td>
<td>Exchange stack top with HL</td>
<td>HL ← (SP)</td>
</tr>
<tr>
<td></td>
<td>SP</td>
<td>1</td>
<td>SPHL</td>
<td></td>
<td>Move HL to SP</td>
<td>SP ← HL</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>2</td>
<td>IN</td>
<td>pp</td>
<td>Input</td>
<td>A ← (pp)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>OUT</td>
<td>pp</td>
<td>Output</td>
<td>Enable interrupts</td>
<td>Enable interrupts after execution of next instruction.</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>EI</td>
<td></td>
<td>Enable interrupts</td>
<td>Disable interrupts after execution of this instruction.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>DI</td>
<td></td>
<td>Disable interrupts</td>
<td>Stop the processor (may be started again only by interrupt or hardware restart).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>HLT</td>
<td></td>
<td>Halt</td>
<td>No Operation</td>
<td>No operation is performed.</td>
</tr>
</tbody>
</table>

* B, D, or H only
In “Computer Assisted Flight Planning” (March 1979 BYTE, page 206), the author did, indeed, identify a problem suited for the computer. One aspect of the article is unclear and I would like to suggest a solution to the headwind/tailwind limitation described by the author.

The unclear aspect of the article relates to the two pass system. Such a system is not really necessary as the author has already solved the problem. The drift equation is the exact solution of the velocity vector triangle for the crab angle (angle at which the plane must be turned so that the resultant forces of the wind and airplane produce travel in the desired direction). In other words, the drift plus the true course heading yields the true heading to be taken. The given equation is instructive in that all elements are included for students to identify. However, for programmable calculators with very limited program steps, this equation can be shortened by using the identities:

\[ \sin(-x) = -\sin(x) \]
and
\[ \sin(x \pm 180) = -\sin(x) \]

This yields:

\[ \text{drift} = \sin^{-1}\left[ \frac{\text{wind}}{\text{true}} \times \sin(\text{wind-true}) \right] \]

where wind direction is defined as the direction from which the wind is coming.

The author, in solving the velocity vector triangle for the ground speed, uses the Law of Sines. This introduces the limitation that the equation cannot be used for direct tail wind or head wind situations, since a non-physical answer results from division by zero (\( \sin(0) = 0 \)). The use of the Law of Cosines avoids this problem and yields:

\[ GS = \sqrt{(TAS)^2 + W^2 - 2(W)(TAS) \cos(WD - TC - CA)} \]

where:

\( GS \) = ground speed,
\( TAS \) = true air speed,
\( W \) = wind speed,
\( WD \) = wind direction (the direction from which the wind is coming),
\( TC \) = true course heading,
\( CA \) = crab angle.
If you've written computer programs in any language, you must be aware by now what a subroutine is, although you might not have written any. The basic concept of a subroutine is present in all computer languages, although every language implements it a bit differently from the others. In systems based on the 8080, the 8085, or the Z-80, you write CALL SUB to call the subroutine called SUB. On the 6800 and the 6502, it's JSR SUB, while in BASIC it's GOSUB \( a \) where the first statement of the subroutine SUB is on line number \( a \). But regardless of the language, the concept is the same: you have something in your program that you want to do more than once. It may be looking up an element in a table; it may be printing out a list; it may be making an access to a data structure; but whatever it is, you need it at various times in your program. You don't want to have to write out the same instructions over again every time you need that particular job to be done, because this is wasteful of memory space. So, therefore, you group together the instructions that do this job into a subroutine, and then, at any point that you want the job to be done, you put in an instruction to call the subroutine. When the subroutine is finished, it returns to the point immediately following the place where it was called; and this is also done differently in different programming languages — one writes RETURN in BASIC, RET for the 8080 and Z-80, and RTS (return from subroutine) for the 6800 and 6502.

All this is fine if the job you want to do repeatedly is exactly the same every time you want to do it. But, in practice, this is usually not the case. For example, if you are looking up an element in a table, you are probably looking up a different element each time. If you are multiplying two 16 bit quantities — a very common subject for a small system subroutine — the quantities you are multiplying are probably not the same from one multiplication to the next, and the result is also probably not the same variable. This is true even though the logic of multiplication does stay the same. It is this that has led to the idea of subroutine parameters, the subject of this article.

Parameters

In applied mathematics, there is a concept of parameter which will be familiar to those small system users who have backgrounds in engineering or physical science. Consider, for example, the graph of a function. You are usually expressing \( y \) in terms of \( x \), but if you are constructing the graph of a circle, it is sometimes more useful to introduce another variable \( \theta \) to represent the angle, and then to express both \( x \) and \( y \) in terms of \( \theta \). The variable \( \theta \), in this context, is called a parameter. In computer programming, however, whether on large systems or small ones, the word "parameter" has a more general meaning, and one which does not require any knowledge of applied mathematics; it is simply any variable which is used by a subroutine, and which is supplied to that subroutine by the program that calls it.

Parameters of subroutines are related to arguments (sometimes also called parameters or formal parameters) of functions. If you have a function \( f(t) \) or \( g(a, b) \) or \( h(x, y, z) \), then \( t, a, b, x, y, \) and \( z \) are the arguments. On a computer, the value of a function is computed by a subroutine, and this must be considered as one special kind of subroutine. Some languages allow you to use functional notation for functions; thus \( h(x, y, z) \) might be FNH(X, Y, Z) in BASIC, for example (provided that the definition of \( h \) was simple enough). In assembly language, however, one generally uses the same instructions (CALL, JSR, or whatever), whether one is calling a subroutine to calculate the value of a function, or a more general subroutine.

Those who work with big computers have laid out a considerable amount of terminology dealing with parameters and how they are supplied, or passed, to a subroutine by the program that calls it (and sometimes vice versa). One of the purposes of this article is to lay out this terminology for the small system user so that he or she will not have to reinvent the wheel. It should be emphasized that, for a long time, mathematicians be-
lieved that there ought to be a single concept of parameter that would work well in all situations. Gradually we have come to realize that there are at least four, and probably a good deal more, reasonable implementations of parameter passing. These will be detailed in what follows.

**Two Examples**

To illustrate why the concept of parameter differs from one situation to another, let us consider two simple subroutines: an output subroutine and a multiplication subroutine. The output subroutine will be called OUTPUT(X), and its job will be to output the character X. The multiplication subroutine will be called MULT16(I, J, N), and its job will be to multiply the two 16 bit quantities I and J, producing the result N. The problem we are to solve is how to call OUTPUT(Z), OUTPUT(Q), and so on, for various characters we wish to output, and similarly MULT16(A, B, C), MULT16(U, V, W), and so on, for various multiplications we wish to perform.

Consider first the case of the output subroutine. Suppose that in this subroutine there is a variable called X. In order to output Z, for example, we move Z to X just before calling OUTPUT. The same sort of thing will work for Q, or any other character we wish to output. This method of passing parameters is known as call by value. It may be defined more formally as follows. Suppose we have a subroutine such as OUTPUT(X), where X stands for any parameter, such as Z or Q, that might actually be supplied. Here Z and Q are called the actual parameters, and X is called the formal parameter. Then call by value consists of:

1. Moving the value of the actual parameter to the formal parameter. (If there is more than one formal parameter – as in the case of a function \( h(x, y, z) \) – then they must all be moved.)
2. Calling the subroutine.

In assembly language it is very common for X, in a situation such as the above, to be a register. Then all we have to do is to load the register before we call the subroutine; the subroutine assumes that Z, or Q, or whatever stands for X, is in that register. (On the 8080, the Z-80, the 6800, and the 6502, the most common register used for this purpose is the A register, although ISIS, the operating system for the Intellec, which is an 8080 based system, uses the C register.)

If we now look at MULT16, however, we can see without too much trouble that call by value doesn't work. Let us see why not by laying out a specific example. Suppose we are calling MULT16(U, V, W), where MULT16 has been defined as a subroutine with parameters I, J, and N. That is, I, J, and N are the formal parameters, while U, V, and W are the actual parameters. To use call by value, we would first have to move the values of U, V, and W into I, J, and N. That is, U would be moved to I; V would be moved to J; and W would be moved to N. Now we would call the subroutine; and the subroutine, we are assuming, multiplies the 16 bit quantities I and J and sets N equal to the result.

What is wrong with this? Since we were calling MULT16(U, V, W), what we presumably wanted was to multiply the two 16 bit numbers U and V, and set W equal to the result. It is not too hard to see that we did, actually, multiply U by V, because we set I equal to U, and J equal to V, and then we multiplied I by J. But what happens to W? We set N equal to the result of multiplying U by V; but we didn't set W equal to anything. (We also, earlier, set N equal to W – an unneeded and useless operation.) The general situation here is that whenever we have a *formal* parameter that is set to some new value by a subroutine, call by value will not work; the formal parameter will not be set to the new value (or to any new value).

Because of this, people who work with big computers came up with three alternative methods of passing parameters. The first of these is known as call by value and result is a rather straightforward way of fixing the bug in call by value that should be evident from the preceding discussion. In fact, what we wanted to do in our MULT16 subroutine was as follows:

1. Set I equal to U and J equal to V.
2. Call the subroutine (which multiplies I by J, giving N).
3. Set W equal to N.

In other words, there are two parameter-passing operations – one just before the subroutine starts, the second one after it ends – and one is the reverse of the other. In the first operation, we move *actual* parameters to *formal* parameters. In the second opera-
tion, we move formal parameters to actual parameters. The parameters we move the first time are the ones that are used by the subroutine; the parameters we move the second time are those that are set by the subroutine.

But how can we tell which parameters are used and which ones are set? It won’t always be the case that the first two are used and the last one is set (if there are three altogether). They might all be used, or two of them might be set, or any number of possible combinations. Again, there is more than one reasonable solution to this problem.

The solution chosen by the designers of a number of computer languages in widespread use by the American military establishment (NELIAC, JOVIAL, CMS-2) was to build the distinction between used and returned parameters into the syntax of the language. In other words, when you call a subroutine in any one of these languages, you would have to specify, in some way, which of these you intended to be used and which you intended to be returned. (JOVIAL, for example, uses a semicolon; we would speak of MULT16(U, V; W), for example, where the semicolon separates the used parameters U and V from the returned parameter W.)

This certainly solves the problem, although only if you are going to use call by value and result, at the cost of making life a trifle more complicated for those who don’t want to have to worry about how parameters are passed.

The other solution, chosen by IBM, is to regard all parameters as both used and returned at all times. This may seem a bit wasteful, but in fact, compared to call by reference (to be described below), it is more efficient, most of the time. It does, however, lead to some strange and unusual results, the most famous of which may be illustrated as follows. Suppose we have a subroutine D(X, Y), where X and Y are the formal parameters, and suppose that this sets X to zero and does not change Y. Now suppose that we call D(L, L). Of course, we would like this to set L equal to zero. But see what happens:

1. Since X and Y are treated as both used and returned, our first step is to set X equal to L and Y equal to L.
2. Now we call the subroutine, which sets X equal to zero and does not change Y.
3. Finally, we return the actual parameters. First we return X by setting L equal to X. Since X is now zero, this will set L equal to zero, which is exactly what we wanted. But now we return Y by setting L equal to Y. Since Y is still the original value of L, this will undo the previous result, and the final outcome will be that L is the same after calling D as it was beforehand!

The behavior illustrated above can be avoided simply by setting L1 equal to L and then calling D(L, L1), rather than D(L, L). In general, when using call by value and result, with all parameters used and returned, one should never use two actual parameters which are the same. The problem above actually happened to a student of this author, who wrote a big FORTRAN program that ran on the CDC 6400, a computer using call by reference — to be described below — but mysteriously failed to run on the IBM 360, a computer using call by value and result. Many hours of analysis traced the bug to subroutine call like D(L, L) above.

**Call by Reference**

Call by reference, historically, preceded call by value and result, although it was not known by that name at that time. The idea of call by reference is to give the subroutine the addresses of its parameters, rather than their values. Then, when the subroutine either uses or sets one of its formal parameters, it does so by making a reference to that address. Let us see how this would work on a small system:

1. On the 8080, you can load the HL register pair with the address of the parameter α with the instruction LXI H, α just before calling the subroutine. Then, in the subroutine, if you need to load this parameter into any register r, you can use MOV r, M; if you need to operate on it arithmetically, you can use ADD M, SUB M, ANA M, and the like; if you need to set it to a new value which is now in register r, you can use MOV M, r. If you need the HL register pair for other purposes in your routine, you can do an XCHG if you don’t need the DE register pair, or you can PUSH H while you use HL and POP H afterward. If there are two parameters, you can load one into HL with LXI H, α as before, and load the other one into BC or DE. If there are several parameters, you can push their addresses onto the stack before calling the subroutine, and pop them back within the subroutine.

2. On the 6800, you can load the X register with the address of the parameter α with the instruction LDX #α (where the # specifies an immediate addressing instruction) just before calling the subroutine. You can now use indexed addressing instructions to manipulate the parameter by loading it (LDAA 0,X or LDAB 0,X), storing it (STAA 0,X or STAB 0,X), or performing arithmetic operations such as ADDA 0,X or ANDB 0,X.
If there is more than one parameter, you can move the addresses of all the actual parameters to fixed locations within the subroutine before calling it. The subroutine can then load each of these into the X register when needed, after which any of the indexed instructions discussed above may be used.

3. On the 6502, there is a general method involving loading the X register, just before calling the subroutine, with the address of a table of addresses of actual parameters. That is, we execute LDX #α where we have written (in page zero):

```
α DFB U MOD 256
DFB U/256
DFB V MOD 256
DFB V/256
DFB W MOD 256
DFB W/256
```

for example, defining a byte for the low order address and then for the high order address of each of the parameters U, V, and W. One can then make reference to the actual parameters by indexed indirect addressing: LDA (0,X) for U, LDA (2,X) for V, and LDA (4,X) for W. This is perfectly general, since LDA (load) can be replaced by STA (store), ADC (add with carry), CMP (compare), AND, and so on.

4. On the Z-80, you can (as always) mimic the 8080, or you can use registers IX and IY to contain the addresses of parameters.

An additional advantage of call by reference is that it allows you to have, as a parameter, the name of an array. For example, you might be writing a subroutine to compare two character strings to see if they are the same. There would be two parameters, namely the two character strings. If you used call by value, you would have to move these entire strings into new locations just before calling the subroutine. This would be wasteful of both time and space, and is, in fact, never done; even systems that use call by value or call by value and result, if they allow array names as parameters, use call by reference (or call by name, to be discussed below) for these. Thus you would only be passing, from the program to the subroutine, the two string starting addresses; that is, for each string, the address of its first byte.

One important source of confusion, when call by reference is used, has to do with how to return a parameter. A large number of programmers try, when they are writing a subroutine, to have it put its answer “somewhere” and then furnish the main program with the address of where that “somewhere” is. This never works, because the main program has no way of using that information. It is not up to the subroutine to tell the main program where the information is to be returned; it is up to the main program to tell the subroutine where to return the information, and then the subroutine must return the information to that point. In particular, the subroutine will never be right if it returns a parameter to a fixed location. When writing a subroutine, if call by reference is used, it should be remembered that this subroutine can be called more than once, with different actual parameters each time, and therefore, when it changes the value of one of its actual parameters, that change must be made by storing this new value in an indexed location – where the index is normally the HL register pair on the 8080, the X register on the 6800 and 6502, and (possibly) the IX or IY register on the Z-80.

Call by reference is, in general, more inefficient than call by value and result, particularly if we make reference to a parameter inside a loop. One technique that has been tried on big computers, and works rather well for subroutines that take large amounts of time, is address modification. This involves storing the addresses which are passed as parameters directly into the instructions that use them. Unfortunately, this technique is inappropriate in most microcomputer systems, where the instructions are in read only memory and thus cannot be modified as the program is running. It should also be mentioned that on some systems which use both call by reference and call by value and result, the second of these is implemented as a special case of the first. That is, it is always the addresses, or references, that are passed (so that there is only one kind of standard subroutine protocol rather than two), but, whenever call by value and result is to be used, the subroutine – rather than the main program – performs the setting of formal parameters to actual parameter values and vice versa.

Call by Name

This brings us, finally, to call by name – the easiest to define, and yet the hardest to understand, of the better known parameter passing methods. For years, call by name was a pons asinorum among big computer software people; that is, the way of distinguishing the bright from the dumb, or the “with-it” from the “not-with-it,” was whether you understood call by name. Lately there has been a bit less interest in call by name among practical computer people, since, although it was used in ALGOL 60, one of the first big computer
languages (in both senses – big [computer languages] and [big computer] languages), it has not been used in most languages developed since then. But an understanding of it, and of some of the problems that arise with it, is still essential to the amateur as well as the professional computer scientist.

Call by name is defined as follows. Suppose I have a subroutine with a formal parameter X. Suppose I call this subroutine, with actual parameter Y. Then call by name implies that the subroutine is executed as if we had gone through it and substituted Y for every occurrence of X.

There is one important proviso to the above, which may be illustrated as follows. Suppose that in the subroutine we have A = B·X. Suppose now that the actual parameter is not Y, but rather U+V. (It is quite permissible to call SUB(U+V), for example, where SUB is the name of a subroutine.) Now we would like to proceed as if A = B·X really means A = B·(U+V); but if we substitute U+V for X, as in the above definition, we obtain A = B·U+V, which is not quite the same. Therefore we need to change the definition so as to specify the insertion of parentheses. On the other hand, it should also be clear that we do not want to insert parentheses all the time. For example, the variable A could have been the formal parameter, rather than X. In this case, the actual parameter could not be U+V, because then A = B·X would be interpreted as (U+V) = B·X, which makes no sense. But suppose the actual parameter is Y, just as before; we still don't want to write (Y) = B·X (with parentheses) in BASIC or any other algebraic language. Therefore the rule is that the actual parameter is substituted for the formal parameter, *inserting parentheses wherever syntactically possible* (this is the phrase used in the definition of ALGOL 60).

So long as the actual parameters are not expressions like U+V (or like A(I), which could be either a subscripted variable or a reference to a function), call by name is almost identical to call by reference. Therefore, in studying the differences between the two, we have to look at the general rules for handling actual parameters which are expressions. These are that an actual parameter cannot be an expression (other than a single variable, either subscripted or not) when the corresponding formal parameter is returned, as we have illustrated above with the formal parameter A and the actual parameter U+V; and, of course, a formal parameter can never be an expression.

Suppose now that in our subroutine we have S = S+X, where X is a formal parameter, and the corresponding actual parameter is A(I). (This is a simplification of an actual example given with the definition of ALGOL 60.) Therefore S = S+X becomes S = S+A(I). But now suppose that we want to do this for I = 1 to 10. That would be, presumably, a way of adding the numbers A(1) through A(10), if S were originally set to zero. If we use call by reference, however, this won't work. In call by reference, the address of the actual parameter – in this case, the address of A(I) – would be given to the subroutine. When the subroutine does S = S+X, it would get X from the location which has that address. But that location is a *constant* location – the location, in fact, of A(I) where the variable I has whatever value it had before the subroutine was called. This means that we add X ten times, whatever X is, and in this case we add the *same* value of A(I) ten times, rather than adding A(1) through A(10).

How would we implement call by name? In the above case, when the subroutine does S = S+X, it has to have a way of finding out whether X will stand for a different variable each time. Therefore it loads S and then calls a subroutine to find the value of X, which it then adds and stores the result in S. This means that it is *the address of the start of this subroutine* that is passed, rather than the address of X itself as in call by reference. (This is known as Jensen's device, after a programmer at Regnecentralen, or the National Computer Center of Denmark, who used it in implementing ALGOL 60.) We should remark that there is another entirely different way of implementing call by name, which is to replace each *call* to a subroutine, separately, by the subroutine with the substitutions performed as discussed above. This won't work for ALGOL 60, because it won't work, in general, for recursive subroutines, and it also takes up quite a bit of space if the subroutines are long.

Call by name is considerably less efficient than the other methods we have discussed, which is a big reason for its general decline. Nevertheless, it has its own unexpected advantages. Let us consider a subroutine like D(X, Y), which we discussed earlier, but this time suppose that it simply uses X and does not use Y, and let us call D(A, F(B)), where F(B) is a reference to a function. Suppose further that the calculation of F(B) (for some reason) gets the computer into an endless loop. If we use call by name, then, since we never use Y, we have no occasion to call the subroutine that calculates Y – that is, we never call F(B). If we use call by value, however, the first thing we do is to set X equal to A and Y equal to F(B). The result is that we get into the endless loop, in this case, if we use call by value, but not if we use call by name.
A "Tiny" Pascal Source Creator

Thomas W Phillips MD
RD 1-551
Chenango Lake Rd
Norwich NY 13815

I would like to thank you for publishing "A 'Tiny' Pascal Compiler" by Kin-Man Chung and Herbert Yuen (September, October and November 1978 BYTE). I now have the compiler working without problems on my 8080 system (Altair with North Star Disk). It is a fascinating way to learn about compilers and Pascal. I am not sure how Mr Chung and Mr Yuen create their Pascal source programs; however, listing 1 shows my method of creating the Pascal source. Editing is easy with North Star BASIC.

Listing 1

100 DATA "CALCULATE SQUARE."
110 DATA "VAR A,B:INTEGER;"
120 DATA "BEGIN"
130 DATA "READ (B#);"
140 DATA "A = B*B;"
150 DATA "WRITE (13,10);"
160 DATA "WRITE (B#,'SQUARED IS',A#);"
170 DATA "END."
180 DATA "END."
190 DATA ""
200 DATA "XX"
210 DIM A$(100)
220 A$="XX"
230 OPEN #1,"A"
240 READ A$
250 IF A$(1,2)="XX" THEN 290
260 WRITE #1,A$,NOENDMARK
270 A$=""
280 GOTO 240
290 CLOSE #1
300 CHAIN "PASCAL"
READY
RUN

PASCAL & PASCAL 2
P-CODES STARTS AT 0000

0 $A
0 ! CALCULATE SQUARE.
0 VAR A,B:INTEGER;
1 BEGIN
1 READ (B#);

Listing 1 continued on page 232
Circle 78 on inquiry card.

$9.95 per pack. More sound programs coming: TRS-80 and Compucolor, too!

FOR PET. TRS 80, COMPUCOLOR.

Level II. $39.95 For Compucolor II (includes Diskette).

SOUNDWARE SOFTWARE FOR PET 8K PET!

Compatible with all CB 2 sound devices. Features sound, super graphics, instruction booklet. 90 day warranty.

1. ACTION PACK—Breakthru (8 versions) / Target / Caterpillar
2. THE CLASSICS—Checkers (8 versions) / Backgammon / Piano Player
3. WORD FUN—Speller (4 versions) / Scramble / Flashcard

$9.95 per pack. More sound programs coming: TRS-80 and Compucolor, too.

To Order: Send to CAP Electronics, Dept. B, 1884 Shulman Ave., San Jose, CA 95124, or call (408) 371-4120 VIS A Master Charge accepted. No charge for shipping when payment is included. Please add 15% for C.O.D. Calif. residents add 6% tax.

Prices subject to change without notice.

DEALER & DISTRIBUTOR INQUIRIES WELCOME.

PROCONSUL
A COMPUTER SIMULATION GAME

A long time ago, in a galaxy far, far away, there was a planet... This planet was very much like the earth at the time of Julius Caesar: A time of some technical ability, but more a time of vaunting ambition; a time of desiring not only to see the other side of the hill, but to conquer it as well.

There are 210 provinces on the PROCONSUL world of various sizes, types and resources. Each player starts with four provinces and vies with the other players, the weather, the possibility of revolt or plague in building an economy and an empire. The goal? To conquer more than half of the provinces on the PROCONSUL world.

PROCONSUL is a true simulation game. Each player has control over farmers, artisans and basic workers and can assign them to various tasks, assign them to another labor category, move them to other provinces or draft them into the military. The player can manipulate the production and distribution of the commodities produced by the labor units. In a battle not only are the relative sizes of the opposing forces taken into consideration, but the morale and experience of each army as well.

PROCONSUL is $1.50 per turn, with turns every three to four weeks. Cost to enter a PROCONSUL game is $1.00, which includes a setup fee, the first four turns, and a copy of the rules. Individual copies of the rules are available for $1.00, which will be credited to anyone who then decides to play in a game. Games are starting all the time and there are no long delays about getting into one.

For further information, or to join a game write to:

NIX OLYMPICA
PROCONSUL
P.O. Box 33306
Phoenix, AZ 85067
Digital Introduces LSI-11/23 and PDP-11/23 Microcomputers

This new microcomputer called the LSI-11/23 from Digital Equipment Corp, 146 Main St, Maynard MA 01754 has the functionality and software compatibility of a midrange minicomputer, yet it consists of two 5.2 by 8.9 inch (13.42 by 22.61 cm) boards and backplane. A rack mountable, packaged version, the PDP-11/23, has also been announced. Both versions can run the RSX-11M and 11S operating systems that were previously available only on mid to high-range PDP-11 minicomputers.

The LSI-11/23 features 256 K bytes of memory capacity. It uses the full instruction set of the PDP-11/34 minicomputer, and software supported memory segmentation and protection features of the RSX-11M and 11S multitasking, multiuser operating systems. The LSI-11/23 has the same small size circuit boards as the LSI-11/2, which permit easier placement in instruments and specialized systems. The LSI-11/23 has an optional floating point processor integrated circuit.

Besides accommodating RSX-11M and 11S software, the LSI-11/23 and PDP-11/23 run all software developed for the LSI-11 family without modification. This includes the RT-11 operating system and high level languages such as BASIC, FORTRAN IV and FOCAL. The LSI-11/23 is at least twice as fast as previous LSI-11 family members.

The system is plug compatible with the entry level LSI-11/2. It is also software compatible with the LSI-11/2 and PDP-11 minicomputers.

In 100 unit quantities, the LSI-11/23 and PDP-11/23 are priced at $1,758 and $4,500 respectively. The single unit price of the PDP-11/23 is $6,800. A new programmable read only memory board for $300 and programmable read only memory programmer for $1,975 have also been introduced.

Circle 623 on inquiry card.

Design, Build and Test Circuits With BredBord Kits

These BredBord kits are prototype design aids developed to simplify circuit design and building, and testing and wiring without patch cords or solder. The 8 BredBord kit models are assembled on heavy duty, glass epoxy boards, copper clad on each underside for ground plane. They feature color coded, insulated binding posts and rubber mounting feet; and a designated number of test points and capacities for multi-pin dual-in-line package integrated circuits. Each of the kits is made to interface with meters, scopes and other devices. The packages are produced in three types to accept different BredBord sizes. For further information, contact Herman H Smith Inc, 812 Snediker Av, Brooklyn NY 11207.

Circle 624 on inquiry card.
This book is a good introduction to computing and the BASIC language for anyone, regardless of their background. David Simon assumes you have no previous programming experience and no familiarity with computing concepts. From the ground up, he builds on the fundamental concepts and teaches each part of the BASIC language in turn. Examples illustrate each step and six fully solved problems are included. This is the “beginner’s book” in BASIC we at BITS have been looking for.

THE BASIC WORKBOOK: CREATIVE TECHNIQUES FOR BEGINNING PROGRAMMERS by Kenneth Schuman, Jr.

This book contains lecture notes, exercises and problems for people learning BASIC. In a hands-on workbook style, Kenneth Schuman covers statements, loops, functions, variables, input/output and strings. Simulation and plotting are introduced. The examples and problems are runnable in virtually any version of BASIC. If you’re learning BASIC, complement your studies with this book!

117 pp. $5.95

BASIC WITH STYLE: PROGRAMMING PROVERBS by Paul Nagln and Henry Legard.

“Programmers can and should write programs that work the first time.” This statement may sound idealistic to those accustomed to long hours of debugging. Yet, it’s the theme of this book—a unique collection of “proverbs” or rules and guidelines for writing more accurate error-free programs. Newly rewritten, this book now emphasizes structured programming and all examples are in BASIC.

134 pp. $5.95

BASIC BASIC, 2ND EDITION by James S. Coan.

If you're not already familiar with BASIC, James Coan's BASIC BASIC is one of the best ways to learn about this popular computer language. BASIC (which stands for Beginner's All-Purpose Symbolic Instruction Code) is easy to learn and easy to apply to many problems. BASIC gives you step-by-step instructions for using a terminal, writing programs, using loops and if-then-else, solving mathematical problems, understanding matrices and more. The book contains a wealth of illustrations and example programs, and is suitable for beginners on many different levels. It makes a fine reference for the experienced programmer, too.

256 pp. $8.95

ADVANCED BASIC by James S. Coan.

Advanced BASIC is the companion volume to James Coan's Basic BASIC. In this book you'll learn about some of the more advanced techniques for programming in BASIC, including string manipulation, the use of files, plotting on a terminal, simulation and games, advanced mathematical applications and more. Many useful algorithms are covered, including some clever sorting techniques designed to reduce program execution time. As with Basic BASIC there are many illustrative example programs included. BASIC doesn't have to be basic with Advanced BASIC!

184 pp. $8.95

BASIC AND THE PERSONAL COMPUTER by Dywer/Chittfield.

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 6 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

THE LITTLE BOOK OF BASIC STYLE by John Nevison.

Structure, style, correctness, maintainability. Attributes of good programming are getting much attention, and well they should. When one considers what we invest in programs, their manageability and efficiency become very important. Here these concepts are explained, along with 19 rules and many examples in BASIC to help improve your programming style.

151 pp. $5.95

SOME COMMON BASIC PROGRAMS by Jon Poole and Mary Borchers, published by Osborne and Associates.

At last, a single source for all those hard to find mathematics programs! Some Common BASIC Programs combines a diversity of practical algorithms in one book: matrix multiplication, regression analysis, principal components, Simpson's rule, root of equations, operations on vectors, chi-square test, check writer, geometric mean and variation, coordinate transformation and a function plotting algorithm. These are just some of the many programs included. For only $8.50 you can buy the kind of programs previously available only as part of software math package systems for large scale computers. All the programs are written in a restricted BASIC, suitable for most microcomputer BASIC packages, and have been tested and debugged by the authors.

192 pp. $8.50

BASIC WITH BUSINESS APPLICATIONS by Richard W. Lott.

This book focuses on the BASIC language and its application to specific business problems. The book is divided into two sections. Part one introduces the BASIC language and the concept of logical flowcharting. Part two presents problems and possible solutions. Topics include: interest rate calculation, break-even analysis, loan rates, and depreciation. Exercises at the end of each chapter give a greater understanding of BASIC by actual programming. This book is a great aid to the beginner wanting to learn BASIC without having a technical or scientific background.

284 pp. $11.50

BASIC COMPUTER GAMES: MICROCOMPUTER EDITION edited by David H. Ahf.

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.

* 188 pp. $7.50

GAME PLAYING WITH BASIC, by Donald D. Spencer.

You'll enjoy the challenge of competing with your own computer. Games described include: 3-D Tic Tac Toe, Nim, Roulette, Slot Machines, Magic Squares, Keno, Morra, Baccarat, Knight's Magic Tour, and many others. The style is non-technical, and each section gives complete rules for the game, how it works, illustrative flowcharts, and example outputs for each program.

The last chapter contains 26 games for reader solution, including Hexapawn and Poker Dice.

166 pp. $7.95
THE FIRST TO OFFER PRIME PRODUCTS TO THE HOBBYIST AT FAIR PRICES NOW LOWERS PRICES EVEN FURTHER!

1. Proven Quality
Factory tested products only, no retests or failures. Guaranteed money back. We stand behind our products.

1979 CATALOG NOW AVAILABLE.
Send $1.00 for your copy of the complete catalog of computer products. A must for serious computer user.

Circle 4 on inquiry card.
SOFTWARE

Key-to-Disk Software Available for Microcomputers

A new key-to-disk software (P1-KTDS), designed to run on 8080 and Z-80 microcomputers, has been announced by Phone 1, POB 1522, Rockford IL 61110. The software supports four video terminals, four floppy disk drives, line printer, and 3780 communications. The P1-KTDS package allows up to four video users to define and select as many as four screen formats per user. Each format may contain as many as 40 user specified fields. Constant data fields may also be specified. In addition to data entry, data verification is also included as a feature of the system. Verification is done on each field specified as a verify field whenever the verify option is enabled. Each of the four users has a separate disk drive that stores the formats and data records for the particular video assigned to the drive.

The P1-KTDS software is currently shipping on Phone 1's P1-S Data Concentrator product which utilizes the 8080 microcomputer, 8214 and 8259 interrupt controllers, 8251 terminal and printer controllers. Mylar decals, which attach to the front edge of the video keytops, guide the user in efficiently entering data.

P1-KTDS source module on CP/M or FDOS III compatible disk, limited use license, users manual, and four sets of keyboard decals are priced at $2,500. The users manual is available separately for $35.

Business Software Packages

A line of 21 fully integrated and auto-chaining business software packages is available from Univair International, 10327 Lambert International Airport, St Louis MO 63145. Some of the major programs include General Ledger, Accounts Payable, Accounts Receivable, Payroll, Inventory, Dental Management System, Medical Management System, Real Estate Multi-List, Insurance Agency, Credit Union, Data Base Management, and Word Processing. All programs are run under CP/M or IMDOSt with commercial BASIC and 31 K bytes of programmable memory. A system of automatic chaining, posting, file backups, and updates is incorporated.

The cost of each program on an eight inch soft-sectored floppy disk is $395. Complete source code and operators manual are provided. Programs are also available on five inch Northstar or five inch Micropolis disks.

Software for the Apple II Computer

Softape, 10756 Vanowen St, North Hollywood CA 91605 has an extensive selection of software available for the Apple II computer. One such program, AppleTalker accepts voice or audio information through the cassette input port, digitizes the information, and stores it in numbered tables in the computer's memory. The stored information can then be played back using the Apple's on board speaker.

Apple-Lis'ner allows the user to communicate with the Apple II computer via spoken words. By using a cassette recorder and microphone, Apple Lis'ner will listen for the words or phrases it has learned and respond under program control.

For more information on these and other Apple II programs, write to Softape at the above address.

The Wordsmith is priced at $299. For a Micropolis disk system, the Wordsmith is priced at $399. It stores its data in numbered tables.

8080 Simulator for the 6502

Now available in a KIM-1 version, the 8080 Simulator for the 6502 processor executes the entire 8080 instruction set. All internal 8080 registers are maintained, ready for convenient examination and modification of their contents. In its minimum configuration on the KIM-1, the 8080 Simulator supports register single step, program counter single step and run modes. It also offers an input and an output port, breakpoint operation, and rejection of illegal op-codes.

The 8080 Simulator runs in less than 1 K of memory, leaving up to 224 bytes of 8080 programming space on an unexpanded KIM-1. The Simulator may be relocated in read only memory and can be adapted to other 6502 based systems.

Well suited to all but time sensitive applications, the 8080 Simulator may be used to assist In the design and testing of 8080 software, used as a training aid or used for running most 8080 application software. The package consists of a KIM-1 format cassette tape, a user manual and a complete, commented assembly level source and object listing. The price is $18 plus $1.50 for postage and handling. For further information, contact Dann McCreary, 4751 Mansfield St #2B, San Diego CA 92116.

Software for the Microcopolis Floppy Disk System

The Basically Speaking Co has announced the availability of software for the Microcopolis five inch floppy disk systems. Statpak includes the ability to create a data file and to do multiple statistical analyses on a data base. Available statistical functions include Chi-Square, analysis of variance and linear regression.

Gradebook allows school teachers to use their computer as a gradebook. Multiple classes are allowed, as well as missing assignments, excused absences, and addition and deletion of student records. A grade figuring program called Reportcard is included.

Word Processing System for Z-80 Based Computers with North Star Disk

The Wordsmith is a word processor for Z-80 based computers with North Star floppy disk systems, an RS-232 terminal and a Diablo 1620 or equivalent printer. It provides complete cursor control, block movements, string searches and alterations, insertion and deletion of text, and other editing functions through the use of control commands.

Print formatting commands are entered along with the text and allow the format to be changed while the printing is taking place. The format commands include right justification, setting of margins, automatic paging and headers, four types of paragraphs, insertion of variable data into the text, and operator instructions. Form letters are easily produced, each one personalized for the recipient, through the use of simple text commands. Disk file creation, deletion and updating are handled automatically.

The Wordsmith is priced at $299. For further information contact Southwest Micro-Systems, POB 20088, Riverside CA 92516.
Assembled and Tested
Added at Ithaca Audio

Field-proven reliable engineering

Over 15,000 boards worldwide prove Ithaca Audio provides the quality and reliability you demand. Ithaca Audio Boards are fully S-100 compatible, featuring gold edge connectors and plated-through holes. All boards (except the Protoboard) have fully buffered data and address lines, DIP switch addressing, solder mask and parts legend.

- **Z-80 CPU Board** still the most powerful 8-bit central processor available. Featuring power-on-jump, provision for on-board 2708. Accepts most 8080 software.
  - A&T 4 mHz $205.00
  - A&T 2 mHz $175.00
  - Blank PC  $35.00
- **Disk Controller Board** controls up to 4 single or double sided drives. Supported by a host of reliable software packages: K2 FDOS, Pascal, Basic and complete diagnostics.
  - K2 Disk  $75.00
- **Video Display Board** features the full 128 upper/lower case ASCII character set. Easy-to-read 16 line x 64 character format can be displayed on an inexpensive video monitor or modified TV set. Includes TTY software. Add our powerful 8K FDOS to create a versatile operator's console.
  - A&T  $145.00
  - Blank PC  $25.00
- **8K Static RAM Board** high speed static memory at a reasonable cost per bit. Includes memory protect/unprotect and selectable wait states.
  - A&T 250 ns $195.00
  - A&T 450 ns $165.00
  - Blank PC  $25.00
- **2708/2716 EPROM Board** indispensable for storing dedicated programs and often used software. Accept up to 16K of 2708's or 32K of 2716's.
  - A&T (less EPROMs)  $95.00
  - Blank PC  $25.00
  - 2708 EPROMs  $11.00

**8” Disk Drives**
Shugart compatible Memorex 550’s are in stock. Single and double density compatible, 330K bytes capacity with your controller or use your own.

Either way  $456

**Memory Prices Tumble**
Ithaca Audio first to break 1€/Byte Barrier

By cutting prices for 32K of RAM to $319 Ithaca Audio becomes the first computer vendor ever to offer high speed memory for less than a penny a byte. Commenting on the announcement, Steve Edelman, Director of Engineering said “Just a few years ago people were wishing for a penny a byte. Now we deliver it.”

Furthermore, the board is available both as a barrier and as a kit, allowing you to save even more.

**Pascal/Z Ready**
The first compiler for the Z80, and the fastest Z80 Pascal ever is now ready. Over one year in development, Ithaca Audio was obviously pleased with the results. “We really are very excited about this.” states Jeff Moskow, Director of Software Engineering, booming over the recently released benchmarks which showed Pascal/Z averaged better than five times the speed of a recent P-code implementation.

“Pseudo-code means a vendor only has to supply one compiler to lots of people using lots of different machines, and that makes his life very easy, but it also means users programs execute significantly slower. Therefore, we chose to write a native compiler that delivers fast re-enterable ROMable code, with no need for an intermediate language and interpreter. That’s where our speed comes from.” As a matter of fact, Pascal/Z is now available as a full 32K memory for most large computers.

More Software:
For those that don’t require the speed of a compiler like Pascal/Z, Ithaca Audio also offers the convenience of BASIC. BASIC/Z, an extended version of TDL’s Super Basic, runs in slightly over 2K and is supplied on an 8” K2 floppy disk. Price including full documentation is $175.00. The Macroassembler is available separately for $50.00. Delivery is from stock.

**How to Order**
Send check or money order, include $2.00 shipping per order. N.Y.S. Residents include tax.

For technical assistance call or write to:

**Ithaca Audio**
P.O. Box 91
Ithaca, New York 14850
Phone: 607/257-0190

---

**Memory Prices**

- 256K A&T $39
- 32K A&T $95
- 32K A&T blank PC $25

**8K FDOS**

- A&T 4 mHz $205.00
- A&T 2 mHz $175.00
- Blank PC  $35.00

---

**Special Offer**

*SAVE $50 when you buy the software as a package*
Microcomputer Offers Pascal in Programmable Read Only Memory

This new microcomputer designed for use with Pascal is being marketed by Control Systems Inc, Drawer EE, Williamsburg VA 23185. The UDS 470 offers Pascal in programmable read only memory as an alternative to assembly language and BASIC for low and medium volume applications where power and fast development are important. They make available a version of UCSD Pascal specifically designed for read only memory and programmable read only memory operation for use in dedicated applications when the development cycle would be slow with assembly language. The Pascal in programmable read only memory feature makes high level programming as easy as assembly language programming. A Pascal program is compiled (instead of assembled) and compiler output (P-code) is burned into programmable read only memory and erased read only memory.

The UDS 470 is a rack mountable system designed for industrial environments (high temperature, vibration, etc). It currently uses the 6800 microprocessor, but can be upgraded to the 6809 or 68000 when they become available. The UCSD system was designed to be machine independent. UCSD's 2.0 version is currently being supplied, but the 3.0 version will be used when UCSD releases it. The standard UDS 470 package includes a processor with 1 K bytes of programmable memory and 2 K bytes of erasable read only memory; serial I/O (input/output) port with automatic reset and VCC monitor; 48 K bytes programmable memory; 16 K bytes erasable read only memory; 5 inch double density floppy disks with interface; 5 V power supply; and a case. The approximate price of the standard UDS 470 is $4000.

Circle 640 on inquiry card.
These Sankyo I/O units are capable of storing and retrieving over 400 characters of data in under two seconds. The flexibility of this device lends itself to numerous applications. As an input reader to a computerized security system, the computer has the ability of identifying the card holder and admitting only those individuals who are authorized to enter the premises during specified time frames. The device is also suitable for maintaining customer information files, or any other application where small amounts of information must be quickly entered into a data processing system. Accepts 3" by 6" ID style mag-cards. (Similar to bank cards.)

**CONNECTORS**

- D-sub male plug & hood
- 90° male plug & hood

**COmponents**

- Apple II
- TRS-80

**SPECIAL CIRCUITS**

- 16k memory (8) 4116's
- 2114 1Kx4 300 $8.95 $8.50 $8.00 $7.50
- 21L02 250nS 1.49 1.19 1.05 .95 .89
- 4045 1Kx4 450 8.95
- 4044 4Kx1 450 5.95 5.50 5.00 4.50
- 4046 4Kx1 450 5.95 5.50 5.00
- 4047 4Kx2 450 5.95 5.50
- 4048 4Kx4 450 5.95 5.50
- 4049 4Kx8 450 5.95 5.50

**MEMORY**

- TRS-80 $65
- Apple II 16k memory (8) 4116's

**PORTABLE DATA ENTRY SYSTEM**

These used data terminals were originally designed for chain store inventory control and order entry systems. The operator enters the inventory control numbers, merchandise on hand and the unit price. After all pertinent data has been entered into the recorder, the main warehouse is telephoned, the handset is placed on the acoustic coupler and the recorded information is transmitted back to the master computer. With a little imagination and one of these portable entry systems, you should be able to exchange programs and computer information with associates across the country.

**FREE MODULATOR**

- Broadcast both analogue and video
- Conversions and stock

**SYSTEM X-10**

- It's not often that California Digital ventures into the distribution of consumer products. However, we have recently acquired a line of X-10 modules, a product that appeals so strongly that we feel we must add it to our product line. This is the X-10 control system, a product that is called the ‘Space Age’. This space age system will control any light or appliance in your home or office. Command signals are transmitted from the control console over your existing wiring. The system X-10 control your stereo, television, or any light fixture on the premises. The basic sampler package comes complete with command console, battery of remote controllers, one each of the appliance modules, lamp module and wall switch. The basic package is priced at only $85.50. Additional modules are available for $35.00 each.

**DigiCast A/V-100**

- R.F. MODULATOR
- Broadcast both Analogue and video
- Conversions and stock

**DigiCast A/V-100**

- Broadcast both Analogue and video
- Conversions and stock

**System X-10**

- It's not often that California Digital ventures into the distribution of consumer products. However, we have recently acquired a line of X-10 modules, a product that appeals so strongly that we feel we must add it to our product line. This is the X-10 control system, a product that is called the ‘Space Age’. This space age system will control any light or appliance in your home or office. Command signals are transmitted from the control console over your existing wiring. The system X-10 control your stereo, television, or any light fixture on the premises. The basic sampler package comes complete with command console, battery of remote controllers, one each of the appliance modules, lamp module and wall switch. The basic package is priced at only $85.50. Additional modules are available for $35.00 each.

**DigiCast A/V-100**

- Broadcast both Analogue and video
- Conversions and stock

**System X-10**

- It's not often that California Digital ventures into the distribution of consumer products. However, we have recently acquired a line of X-10 modules, a product that appeals so strongly that we feel we must add it to our product line. This is the X-10 control system, a product that is called the ‘Space Age’. This space age system will control any light or appliance in your home or office. Command signals are transmitted from the control console over your existing wiring. The system X-10 control your stereo, television, or any light fixture on the premises. The basic sampler package comes complete with command console, battery of remote controllers, one each of the appliance modules, lamp module and wall switch. The basic package is priced at only $85.50. Additional modules are available for $35.00 each.
**Miniature Joystick Catalog**

Measurement Systems Inc announces publication of their new 16 page catalog, *Miniature Joysticks*. These joysticks are used for cursor positioning of many displays including interactive terminals, computer aided drafting, and radar systems. They are also used for mechanism positioning such as microcircuit production equipment and vehicle control. These joysticks are offered for control of one, two, or three axes. They are offered in commercial, industrial and military grades. For further information contact Measurement Systems Inc, 121 Water St, Norwalk CT 06854.

Circle 642 on inquiry card.

**Wintek Corp Offers New Catalog**

This catalog contains a 6800 based single board computer plus 15 support and interface modules on industry standard 4 Y, by 6 Y, 5/8 inch cards for process control and data acquisition. Additionally, 6800 development systems, resident and cross assemblers, and compilers are listed. For further information contact Wintek Corp, 902 N 9th St, Lafayette IN 47904.

Circle 644 on inquiry card.

**The Complete Motorola Microcomputer Data Library**

The Complete Motorola Microcomputer Data Library presents technical data for microcomputer design and implementation. It is divided into three basic segments, each further subdivided into subordinate product categories. The three segments are:

- microcomputer components—microprocessor and microcomputer unit components, together with interface and peripheral components to implement microcomputer systems,
- memory products—basic memory components and add-in and add-on memory subsystems for computer applications,
- microcomputer development systems and subsystems—support products (hardware and software) to design microcomputer systems; board-level subsystems for system implementation.

The organization within each of these basic segments is by device families and application groupings rather than in alphanumeric sequence. Therefore, a comprehensive table of contents provides the reader with a sequential listing of the chapter by chapter content of each segment.

The book is priced at $6. For further information write to Motorola Semiconductor Products Inc, POB 29024, Phoenix AZ 85036.

Circle 645 on inquiry card.

**Buyers Guide Offered Free of Charge**

This buyers guide of microcomputer software, accessories, and supplies is available from Wallace Electronics Inc, 4921 N Sheridan Rd, Peoria IL 61614. Software and accessories for the Apple II and TRS-80, as well as a wide range of computer supplies, are listed on these sheets. The guide is updated weekly. The buyers guide is free of charge, although $.50 should be included to cover postage and handling.

Circle 646 on inquiry card.
**Venus 2001 Video Board**

Assembled and Tested $259.95 • Complete Unit with 4K of Memory and Video Driver on Eprom assembled and tested $339.95

**kit $199.95**

**OPTIONAL:**
- Sockets $10.00
- 2K Memory $30.00
- 4K Memory $60.00
- Video Driver Eprom $20.00
- Text Editor Eprom (Includes Video Driver $75.00)

**S-100 Plug-in • Parallel Keyboard Port**

On board 4K Screen Memory (Optional). On board Eprom (Optional) for Video Driver or Text Editor Software.

**Up and Down Scrolling through Video Memory**

Reverse Video, Blinking Characters.

**Display:** 128 ASC11 Characters 64 X 32 or 32 X 16 Screen format (Jumper Selectable). 7 by 11 Dot Matrix Characters.

---

**ASCII Keyboard Kit $77.**

Assembled and Tested $93.00

- Single +5V Supply
- Full ASCII Set (Upper and Lower Case)
- Parallel Output
- Positive and Negative Strobe
- 2 Key Rollover
- 3 User Definable Keys
- P.C. Board Size: 17-3/16" X 5"
- Control Characters Molded on Key Caps
- Optional Provision for Serial Output

**OPTIONAL:**
- Metal Enclosure $27.50
- Edge Con. $2.00
- Sockets $4.00
- Upper Case Lock Switch $2.50
- Shift Register (For Serial Output) $2.00

**Apple II I/O Board Kit**

Plugs into Slot of Mother Board

- 1 8 Bit Parallel Output Port (Expands 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 $49.00
- 1 Input and 3 Output Ports $64.00

**Dealer Inquiries Invited**
Visible Computer Supply Offers Free General Catalog

Publication Lists 32 BASIC Programs for the PET

32 BASIC Programs for the PET Computer by Tom Rugg and Phil Feldman is precisely that. 32 fully documented programs that are ready to run on an 8 K byte Commodore PET 2001 computer. The reader does have the option of making changes to these programs. This 267 page book covers application, educational, game, graphic display, mathematical and miscellaneous programs. The book is priced at $15.95. For further information, contact Dilitium Press, POB 92, Forest Grove OR 97116. Circle 650 on inquiry card.

Software Magazine Devoted to Radio Shack TRS-80

Owners of Level II Radio Shack computers will appreciate SoftSide, a magazine devoted to providing games and light application software in Level II BASIC. Owners of other personal computers using Microsoft BASIC will also find programs that can be readily converted for their systems.

The particular emphasis of the magazine is simulation games. Readers of recent issues have been able to play football, race a clipper ship around Cape Horn, rule a fifteenth century Italian city-state or chase wild animals on a photographic safari. Light application programs are also published, and have included an income tax program and a personal finance program complete with graphic pictures of checks on the screen. Hints for TRS-80 programmers regularly appear in various places throughout the magazine.

Softside is published monthly and is available by subscription for an annual rate of $15. A special cassette edition which includes the magazine and all the monthly programs in machine readable form is available for $38 for a six month subscription. For further information contact SoftSide Publications, POB 68, Milford NH 03055.

New Book Features Self-Contained Programming Problems

Etudes for Programmers by Charles Wetherell is a collection of large scale problems for learning by doing. Each problem includes a real world background discussion of appropriate programming techniques, detailed requirements for correct solution, extensions, and annotated bibliography. Two of the problems are completely solved by the author. The solutions concentrate on good programming techniques, measuring the quality of the program and the output, and possible extensions of the problem. They are models of what solutions to any programming job should be, and they contain many practical hints about writing good programs.

Additionally, this 200 page book offers references to sources for programming information and further reading about problem subjects. It includes a complete set of four projects for a programming language course: macro-interpreter, compiler, relocating loader and computer simulator.

Etudes for Programmers is priced at $12.95. For further information, contact Prentice-Hall Inc, Englewood Cliffs NJ 07632.

Apple Software Directory

Over 700 software programs for the Apple computer have been compiled into the Apple Software Directory. All programs are listed alphabetically so that the same type of program produced by several sources can be compared. Listings include description, memory requirements, price, format and the source. All sources are listed with addresses.

The directory is printed in two volumes. Volume 1 covers business and utility programs. Volume 2 covers games and entertainment programs. Each is priced at $4.95. For more information, write WIDL Video, 5325 N Lincoln, Chicago IL 60625.
The PET is now a truly sophisticated Business System with the Floppy Disk and Printer which makes an ideal cost efficient business system for most professional and specialized fields. medical, law, research, engineering, education, etc.

**PET 2001**
- 16K $995
- 32K $1295
- 14K ROM Operating System
- 9K RAM Memory
- Video Monitor
- Built in Keyboard
- Digitally controlled tape

**PET BUSINESS SYSTEM**
- Upper Lower Case and Graphics
- Full SLEt Business Keyboard
- Operating system will support multiple Languages (BASIC resident)
- Machine Language Monitor
- 8K ROM Expansion Sockets
- 9 CRT

**BUSINESS SOFTWARE**
- IMSAI
  - The latest solution for all small business problems
  - A wide variety of software available for all your needs
  - Includes dual floppies, 32K RAM, 1024K memory, BASIC, and many other programs

**PET PRINTERS**
- 2023 $849
- 2022 $995
(Includes Tractors)
- Bottom and Rear Tractor Feed
- 8" Paper Width
- 6004 Microprocessor Controlled
- 4K ROM Operating System
- 8K ROM Encoding and Decoding
- 4K RAM
- Uses Single or Dual sided 80 Character Matrix

**DUAL DRIVE FLOPPY DISK**
- $1295
- 300 K Bytes Storage
- High Speed Data Transfer
- Pulls into ISIS Port
- 6004 Microprocessor
- 8K ROM Operating System
- 8K ROM Encoding and Decoding
- 4K RAM

**SINGLE DRIVE FLOPPY UNIT**
- $895
- Intelligent mini floppy
- 117 K net user storage

**PET COMPUTER FACTORY**
- COMMODORE "PET" used only $795
  - 14K ROM Operating System
  - 9K RAM Memory
  - Video Monitor
  - Built in Keyboard
  - Digitally controlled tape

**NEW! PET BUSINESS SYSTEM**
- The PET is now a truly sophisticated Business System with the Floppy Disk and Printer which makes an ideal cost efficient business system for most professional and specialized fields. medical, law, research, engineering, education, etc.

**SPECIAL OFFER**
- **SPECIAL** on Off-Price Printers and Printers
- **PET** COMPUTER FACTORY
- **SPECIAL** on Off-Price Printers and Printers
- **PET** COMPUTER FACTORY
- **SPECIAL** on Off-Price Printers and Printers
- **PET** COMPUTER FACTORY
- **SPECIAL** on Off-Price Printers and Printers
- **PET** COMPUTER FACTORY
- **SPECIAL** on Off-Price Printers and Printers
- **PET** COMPUTER FACTORY
**Answer and Originate Acoustic Coupler**

The AC-312 answer/originate acoustic coupler operates at 300 bps. The device is 103 Western Electric compatible and is switch selectable between originate and answer modes. When in the answer mode, this unit will generate the answer tone necessary to communicate with 300 bps originate only couplers and modems. The AC-312 includes a single plugable printed circuit board for ease of field upgrading to 1200 bps operation and field service. Standard light emitting diode diagnostic indicators are also featured. The AC-312 answer/originate modem is priced at $295. For further information, contact Digicom Data Products Inc, 1440 Koll Circle, Suite 108, San Jose CA 95112.

Circle 653 on inquiry card.

**Real Time Calendar and Clock for Apple II Computer**

Mountain Hardware has announced the Real Time Calendar and Clock for Apple II computers. The Apple Clock keeps time and data in 1 ms increments continuously for over one year. Calendar, clock, and event timer functions are easily accessed from BASIC using routines contained on board read only memory. Some of the features of the Apple Clock include crystal control, on board rechargeable battery to keep the clock running during computer down times; software for calendar and clock routines as well as an event timer contained on board read only memory; and an interrupt feature which can be programmed to make efficient use of computer time. Sample applications include programming a morning printout of appointments; date of transactions; creating games in which elapsed time is important; and time events. The Apple Clock can be added to Mountain Hardware's Introl Remote Control System for real time control and monitoring of remote devices over regular AC wiring.

The price of the Apple Clock is $199 assembled and tested. For further information, contact Mountain Hardware Inc, 300 Harvey W Blvd, Santa Cruz CA 95060.

Circle 654 on inquiry card.

**Interface TRS-80 to Summagraphics' Bit Pad**

Summagraphics Corp has announced the availability of an interface for the company's digitizer, the Bit Pad, which allows connection to the Radio Shack TRS-80 microcomputer. This new interface permits the entry and transfer of X,Y coordinate values for graphics and data entry applications from the Bit Pad to the TRS-80 computer.

The interface is priced at $175, and a cassette containing software is provided. Data is transferred from the Bit Pad in groups of five bytes. The interface is contained in a small separate box that connects to the Bit Pad and the TRS-80. The interface allows use of all other TRS-80 accessories. For further information, contact Summagraphics Corp, 35 Brentwood Ave, Fairfield CT 06430.

Circle 655 on inquiry card.

**Compact Low Cost Alphanumeric Printers**

The DigiTec 6410 and 6420 small desktop printers print 20 columns of alphanumeric characters. Sixty-four different characters are produced in a 5 by 7 dot matrix. The printer can easily replace teletypewriter terminals in applications that don't need 80 column capability. An internal microprocessor makes these new printers reliable and easy to interface. The Model 6410 provides a serial interface to RS-232C and 20 mA current loop systems at 110 bps. The Model 6420 works with 8 bit parallel bus systems at up to 1000 characters per second. They both use the ASCII input format. The single unit price is $395. For further information and special OEM information, contact United Systems Corp, 918 Woodley Rd, Dayton OH 45403.

Circle 656 on inquiry card.
Electrolabs
POB 6721, Stanford, Ca. 94305

FLOPPY SYSTEMS

8" Siemens FDD120-B Drive
All Siemen's options included (in this drive which can be con-
figured hard or soft and single or double density. (Others give
only stripped until) $399.00

“Power One” Model CP206
Floppy Power Unit. For two drives going full-out, and poss-
ibly more on less severe service. 2.6A@24V, 2.6A@6V, 0.6A@8-V.
Beautiful quality. $99.00

DISKETTES (Standard)
8" Boxed 10 $39.00
5¼" Boxed 10 $34.95

Tarbell (“It Works”) Interface
(includes cable set for 2 drives) $265.00 BUT ONLY $219.00
with purchase of two drives.

Cable Kits 10' with 50 cond.
cable and connectors and also
Molex connectors and power
69.
$33.95, and for three drives: $38.95

CABINETS for FDD120 and
801R Drives, or CP206 power
supply. Matte finish in mar-
tack type design. $29.99

Used Sylvania 12" Video Moni-
tors. Composite video 15mhz,
115vac, 50/60hz New Tube. As
shown $109 OEM style without
case. $99, Anti-glare tube option
add $12. Specify p4 or p99

ESAT 200B

BI-LINGUAL 80x24
COMMUNICATING TERMINAL
Scrolling, full cursor, bell,
8x8 matrix, 110-19,200
baud, Dual Font Appli-
cations, Arabic & Hebrew,
Multilingual Data Entry,
Forms Drawing, Music Instruction, Specialized Graphics (e.g. Games, Chemical Plants,
Switchyards) $348.00

We carry keyboards, cases, power supplies, etc., enough to make
an entire system.

NEW CATALOGUE

Crystals
Integrated circuits
Keyboards
Lasers
LSI-11
Media
RAMs
S-100 Components
Z-80 Components

ANNOUNCEMENT!

SUPERDISK

ELECTROLABS is proud to announce appointment as
DISTRIBUTOR by CII-Honeywell Bull.

PRICE BREAKTHROUGH on SUPERDISK 10 MBY! $3495.00
General purpose controller (requires 2 parallel I/O ports) 1500.00
S-100 Controller (DMA) 995.00
“RL-01/RK-05” surrogate 1900.00
(transparent to RT, RS, RX)

SOFTWARE: (CP/M Compatible)
SUPERDOS 1
(Z-80) $695.00
MICRODOS 1
(TRS-80) $199.00
Power supply (switching) $395.00
Enclosure (desktop) $99.00

Removeable Media Cartridge Drive

5¼" MINI-FLOPPY DRIVE
$299.00

— single or double dens-
ity — quick access time
— high reliability &
durability

Mini-floppy CABLE KIT:
for TRS-80 or your
Tarbell controller.
$24.95

Daisy Wheel Printers
Gume Sprint 3/45
New!

Print wheels $8.95
Ribbons $5.95

Print wheels $8.95
Ribbons $5.95

OEM Style mechanism $1399.00

Circle 115 on inquiry card.
What's New?

ASSOCIATIVE COMPUTER MEMORY

Available from Semiconics

Content addressable or associative computer memory is available from Semiconics Associates, 41 Tunnel Rd, Berkeley CA 94705. Called REM (recognition memory), it differs from conventional memory by eliminating serial searching. An item may be accessed simply by being named. REM can be written into and read from like ordinary memory, but has parallel processing functions, including six types of recognize and multiwrite. The recognition operations replace serial searching, while multiwrite allows the processor to write into multiple locations with a single instruction. Individual bit masking may be applied to all of the operations, including ordinary (location accessed) read and write. A data processing system with these functions is known as a CAPP (content addressable parallel processor). Ideal for pattern recognition and information retrieval applications, it is also capable of performing parallel arithmetic operations.

Semiconics' first product is an add-in recognition memory for microcomputers having the 5-100 bus. Called REM 5-100, the board converts the microcomputer to a CAPP by adding new instructions to the instruction set of the processor. The board is organized to make these additional instructions possible without any alteration to the processor.

Recognition memory is organized in 8 bit words and 256 word REM records. It is a static memory with an access time of 200 ns for a single memory location, and recognize or multiwrite time for all REM records of 4 µs. This time does not increase with size of memory. In a system with multiple REM boards, all of these are accessed in parallel during a recognize or multiwrite operation. The REM 5-100 add-in recognition memory board has a capacity of 4 K bytes and is priced at $345.

6800 DEVELOPMENT PAC!

• CONTROL MODULE-single board computer
• 16K RAM MODULE-16K 8-bit bytes
• RS232 INTERFACE-switchable rates
• CASSETTE INTERFACE-300/2400 baud
• FANTOM-II monitor/debug ROM
• EDITOR/ASSEMBLER-on cassette
• CARD RACK, BACK PLANE, POWER SUPPLY
• $695

64K RAM BOARD

The Zg-SYSTEMS 64K RAM board is designed to operate in any 260 based microcomputer having 5-100 bus. It uses 16K dynamic RAM chips, 6 features:

- Board select
- Bank select
- Transparent on-board refresh
- 2 or 4MHz operation (no wait state)
- Memory disable

Compatible with Cromemco system

Fully assembled, burned in, & tested
Available from stock to 60 days
As low as $500.00 in quantities of 100
Price of one

$499.00

PC Board only

$399.00

PLUS FREIGHT CHARGE

Zg SYSTEMS
PO Box 1847, San Diego, CA 92112
(714) 447-3997
SOLID STATE SALES... Announces a Breakthrough in Computer Technology

This remarkable VP-1 computer/interface kit has the following:

**Features**
- It produces composite video output in a 128 x 128 matrix from a direct monitor connection using 8K of memory.
- The system uses a standard 5-100 bus.
- Will operate in computer software when not addressed.
- It displays continuously when not addressed.
- It also produces pseudo-color and graphics (up to 16 grey levels, 4-bit binary).

**Applications**
- Continuous surveillance.
- Inspection of moving parts with proper strobing.
- Visual graphic input to a computer.
- Character or pattern recognition.
- Pictures may be taken directly from a TV without electrical connections.
- The interface kit may be used separately as a 128 x 128 level graphic display.

**Printed Circuit Board**

<table>
<thead>
<tr>
<th>Component</th>
<th>Quantity</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>7Watt-LD-65 Laser Diode 1R</td>
<td>858.50</td>
<td></td>
</tr>
<tr>
<td>7Watt-LD-65 Laser Diode 1R</td>
<td>858.50</td>
<td></td>
</tr>
<tr>
<td>7Watt-LD-65 Laser Diode 1R</td>
<td>858.50</td>
<td></td>
</tr>
<tr>
<td>7Watt-LD-65 Laser Diode 1R</td>
<td>858.50</td>
<td></td>
</tr>
<tr>
<td>7Watt-LD-65 Laser Diode 1R</td>
<td>858.50</td>
<td></td>
</tr>
</tbody>
</table>

**Transistor Specials**

- 74HC33N4 Switching Power: $1.15
- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**Full Wave Bridges**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**Tantalum Capacitors**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**Silicon Power Rectifiers**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**Silicon Solar Cells**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**LIGHT ACCEL. SENSORS**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**LED READOUTS**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**Regulators**

- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**CALL FOR AVAILABILITY & PRICING**

- M74HC1510N CD 1510 Quad Buffer: $1.00
- 74HC4066 CD 4066 Transistor: $0.75
- 74HC74N CD 74HC74 Tri-State: $0.75
- 74HC123 CD 123 Quad Buffer: $1.00
- 74HC4050 CD 4050 Quad Buffer: $1.00
- 74HC4051 CD 4051 Quad Buffer: $1.00

**SOLID STATE SALES**

P.O. Box 746
SOMERVILLE, MASS. 02143 TELEPHONE (617) 747-7093

We ship over 90% of our orders the day we receive them.

Circle 340 on inquiry card.
SOFTWARE

Disk Operating and File Management System for 6800 Microcomputers

INDEX handles both ASCII and binary files, and disk files are automatically created, allocated and de-allocated. Files are referenced by names, and file name parameters are added for name extension, drive number, directory level and a file protection flag.

The INDEX operating system software also features a routine for copying files onto a disk. The console interface segment of INDEX software supports any standard serial ASCII terminal. It features program interrupt for runaway programs; operator start, stop and skip display control; interrupt-serviced, typed-ahead character queue buffer; and a secondary line editing queue buffer.

INDEX versions are available for the PerCom LFD-400, Southwest Technical Products' MF-68, Smoke Signal Broadcasting Company's BFD-68 disk systems, and Motorola's EXORciser development system.

INDEX is supplied on two 5 inch disks together with a users manual for $99.95. For further information contact PerCom Data Co, 318 Barnes, Garland TX 75042.

New Self-Merchandising Software From ComputerLand

ComputerLand is now making personal computing software available through all participating ComputerLand stores. SoftSpot is a custom designed, self-merchandising fixture offering off-the-shelf programs for personal finance, time budgeting, education, games, stock analysis, stock portfolio evaluation, and more. SoftSpot programs start at $7.95.

Customer education about personal computers and their applications are offered through MainBrain. Books, self-study cassettes and video tapes, and “in person” lecture programs are available from well known publishers. MainBrain has self-service instructions to assist customers in making their own choice of multimedia educational products.

For further information contact ComputerLand, 14400 Catalina St, San Leandro CA 94577.

PET SPECIALS

<table>
<thead>
<tr>
<th>PET</th>
<th>LIST</th>
<th>SALE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PET 16N</td>
<td></td>
<td>$95</td>
</tr>
<tr>
<td>PET 16B</td>
<td></td>
<td>$95</td>
</tr>
<tr>
<td>PET 32N</td>
<td></td>
<td>$1295</td>
</tr>
<tr>
<td>PET 32B</td>
<td></td>
<td>$1295</td>
</tr>
<tr>
<td>PET 16S</td>
<td></td>
<td>$95</td>
</tr>
<tr>
<td>PET 32S</td>
<td></td>
<td>$1295</td>
</tr>
<tr>
<td>PET 8K</td>
<td></td>
<td>$795</td>
</tr>
</tbody>
</table>

ASK ABOUT EDUCATIONAL DISCOUNTS ON PET

- IEEE - RS232 Printer Adaptor for PET $59.50
- BETSI PET to S-100 Interface & Motherboard $119.00
- PET Connectors - Memory (8K) $2.95
- Parallel or IEEE $2.25
- Cassette Port $1.80
- Personal Information Management System—Scelbi $8.90
- Protect-A-Pet dust cover $9.95
- EXS 100 Floppy Disk Controller for PET $299.00
- PET 6550 RAM $16.20
- Microchess for PET (Peter Jennings) $17.90
- PET 4 Voice Music Board (MTUK-1002-2) $48.00
- Music Software (K-1002-3C) for PET $19.00
- Cmc Word Processor program for PET $25.00
- Bridge Challenger program for PET $13.50
- Play and reply bridge hands against the PET $13.50
- Graphics Utility Package for PET $13.50
- Stimulating Simulations $13.50
- Kite Fight - Michael Riley $7.95

Minimum Order $10.00

WRITE FOR 6502 AND S-100 PRODUCT LIST

A B Computers

115 E. Stump Road
Montgomeryville, PA 18936
(215) 699-9336

Circle 1 on inquiry card.
RCA Cosmac Super Elf Computer

Compare features before you decide to buy any digital computer. There is no other computer on the market today that has all the desirable benefits of the Super Elf for 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 observe microprocessor chip operation with the unique Quest address and data displays before, during and after executing instructions. Also, CPU mode and instruction cycles are decoded and displayed on eight LED indicator lamps.

An RCA 1801 video graphics chip allows you to connect to your own TV with an inexpensive video modulator to display graphics. There is a speaker system included for playing 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 decimal,残忍, real, reset, memory plus.

Super Expansion Board with

This is truly an astounding value! The 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 protection, memory lockout and resolution. Provisions have been made for all other options on the same board and if it is really into the hardwired cabinet along with the Super Elf. The board includes slots for up to 6 of EPROM (2708, 2732, 2716 or TI 2716) or 27C025 high speed EPROMs can be used for the monitor and Tiny Basic or other purposes.

A 1K SUPER ROM Monitor $19.95 is available as an option on the Super Expansion Board which has been preprogrammed with a program loader editor and error checking multi line cassette read-write software. (Replaceable cassette file) is another exclusive from Quest. It includes registers, scan readout, block move capability and video graphics driver with blinking cursor. Break points can be used with the register save feature to observe program flow, then follow with single step. The Super Monitor is written with subroutines allowing users to take advantage of monitor functions simply by calling them up.

Auto Clock Kit $15.95

DC clock with 4-30 displays. Uses National MA-1012 module with alarm option. Includes light dimmer, crystal timebase PC boards. Fully regulated power supply. Add $3.95 for beautiful dark gray case. Best value anywhere

RCA Cosmac VIP Kit $229.00

Video computer with games and graphics fully assembled and tested. $249.00

Not a Cheap Clock Kit $14.95

Includes everything except case 3 PC boards. 6-50 LED Displays, .30 clock chip, transformer, all components and full instructions. Orange displays also available. Same kit w/AC displays. Red only. $21.95 Case $11.75

60 Hz Crystal Time Base Kit $4.40

Converts digital clocks from AC line frequency to crystal time base. Outstanding accuracy. Kit includes 60 Hz crystal, resister, capacitors and trimmer.

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.

Digital Temperature Meter Kit


NiCad Battery Filler/Charger Kit

Opens shorted cells that won't hold a charge and then charges them up, all in one kit. Full parts and instructions. $27.25

PROM Eraser Will erase 25 PROMs in 15 minutes. Ultraviolet, assembled $34.50

Rockwell AIM 65 Computer

6502 based single board with full ASCII keyboard and 20 color thermal printer. 20 color alphanumeric displays, ROM monitor, fully expandable. $375.00. 4K version $450.00. 4K Assembly manual $150.00. Power supply assembled in case $50.00.

INTEGRATED CIRCUITS

-Multi-volt Computer Power Supply

8v 5 amp, 18v 5 amp, 5v 5.5amp, 5v 12v, 12v 2 amp, 15v option 9v, 12v $19.95 kit with punched frame $34.95. Woodgrain case $10.00

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 $37.30 Kit less case $47.75 Precise to 15 MHz. $19.95

79 IC Update Master Manual $35.00


Same day shipment. First line parts only. Factory tested. Guaranteed money back. Quality ICs and other components at factory prices.

Stopwatch Kit $26.95

Full 6 digit digital battery operated, 2 volt cells. 3.768 MHz crystal accuracy. Times to 59 min, 59 sec, 59.9/1000 sec. Times add, split and Taylor. 7200 chip, all components minus case. Full instructions.

Hickok 3" Digital LCD Multimeter

Battery operated. 10 ranges. 0.5% or better. Resistance 6 low power ranges 0-1000 0m2 digital D.C current 0-100mA D.C voltage 0-100V. Hand held, LCD displays, auto zero, polarity, over range. $74.95

S-100 Computer Boards

8K Static RAM Board $139.00 1K Static RAM $26.00 2K Static RAM $30.00 4K Static RAM $42.00 8K Static RAM $50.00 64K Dynamic RAM $70.00 8K/16K Eprom Kit (less PROMS) $89.00 Video Interface $139.00 Masterboard $39. Expand Board $89.00

FREE Send for your copy of our NEW 1979 QUEST CATALOG. Include 5c stamp.

P.O. Box 4430X Santa Clara, CA 95054
For will call only (408) 988-1640
2322 Walsh Ave.
Transistor Checker

- Completely assembled -
- Battery Operated -

The AA Transistor Checker is capable of checking a wide range of transistor types, either in circuits or out of circuit. To operate, simply plug the transistor to be checked into the front panel socket, or connect it with the alligator clips. The unit safety and automatically identifies NPN and PNP types and indicates the voltage and power required for proper operation. The LED display indicates a "Good" or "Bad" condition. (AA battery not included.)

• Trans-Chex $29.95 ea.

Custom Cables & Jumpers

DB 25 Series Cables

Part No. | Cable Length | Connectors | Price
--- | --- | --- | ---
Dip Jumpers

DJ-1 | 1 ft. | 1-14 Pin | $1.95 ea.
DJ-16 | 1-16 Pin | $2.49 ea.
DJ-32 | 1-32 Pin | $2.95 ea.
S6/100 | 1-100 Sizing | $2.95 ea.
S20/100 | 1-200 Sizing | $2.95 ea.

PRINTED CIRCUIT EDGE-CARD

JE730 | $14.95

Microprocessor Components

- 50/100 (100 Spacing) Pins Wire Wrap $6.95
- OJ16.1 16
- OJ24.1 24
- OJ82SP 82
- Bright .357" ht. red display
- Extruded aluminum case (black)
- LEO JE701 wall transformer
- JE2206B

Computer Cassette

- 8 each UPD4161-1 (16K Dynamic RAMs) 250NS
- Documentation for computer

TR-80 16K Conversion Kit

Expand your 4K TR-80 System to 16K. Kit comes complete with:

- 8 each UPD4161-1 (16K Dynamic RAMs) 250NS
- Documentation for conversion

TRS-16K

$75.00

Computer Cassette

- 8 each 15 MINUTE HIGH QUALITY 1-15 Cassettes PLASTIC INCLUDED
- 12 CASSETTE CAPACITY AVAILABLE #0-10-$1.25 #10-$1.50

CAS-6

$14.95

(8 and 6 Cassettes)

Sup 'R' Mod II

UHF Channel 33 TV Interface Kit

Wide Band 5V or Color System Conntetion to PAL for home computer, RGB or CRT monitors, etc.

MOD II is prefitted to Channel 33

$29.95 Kit

Function Generator Kit

- Provides 3 basic waveforms: sine, triangle, square wave
- Output level from 1 to 100,000 Hz
- 6 volt output from 0 to 5 volts (peak to peak)
- Uses a 2 x 47k supply

JE2006B

$19.95

Cassette Controller

The IDEAL FOR TRS 80

- Plug interface to computer, prevents cassette tampering

#CC-100

$29.95

36-Key Unencoded Keyboard

This is a 36-key, terminal keyboard newly manufactured for a large computer manufacturer. It is a separate unit, not attached to any keyboard of PC board. A very solid molded plastic 13 x 8" base suits most applications. IN STOCK $29.95/each

Hexadecimal Unencoded Keyboard

19-key pad includes 1-10 keys, a decimal point, and a shift key.

$10.95/each
HORIZON 16K KIT............$1275.00
16K RAM BOARD KIT .........$250.00
32K RAM BOARD KIT .........$1 475.
VERBATIM DISCS FOR NORTH STAR
BOX OF 10 ............$29. POST PAID

MAXIMUM VALUE FOR YOUR DOLLAR

NORTH STAR COMPUTER PRODUCTS

HORIZON-16K KIT............$1275.00
16K RAM BOARD KIT .........$250.00
32K RAM BOARD KIT .........$1 475.

PHONE ORDERS, 714/633-4460

OR ORDER. VISA, BA, MASTERCHARGE - SEND ACCOUNT NO.,

All ORDERS POSTPAID. U.S. FUNDS. CHECK OR MONEY 

EXPIRATION DATE, INTERBANK NO., & SIGNED ORDER.

CASSETTE AND FLOPPY

Includes 8 TESTED & GUARANTEED M5K 4116 3 16K RAMS.

ORDER. VISA, BA, MASTERCHARGE - SEND ACCOUNT NO.,

All ORDERS POSTPAID. U.S. FUNDS. CHECK OR MONEY

EXPIRATION DATE, INTERBANK NO., & SIGNED ORDER.

American Square
Kivett Dr, Jamestown NC 27282
(919) 983-1056

Circle 5 on inquiry card.

Circle 21 on inquiry card.

Circle 216 on inquiry card.

Circle 314 on inquiry card.

Circle 391 on inquiry card.

Circle 20 on inquiry card.

Circle 19 on inquiry card.
16K EPROM CARD-S 100 BUS$  
$59.95 KIT

OUR BEST SELLING KIT!

USES 2708's!

Thousands of personal and business systems around the world use this board with complete satisfaction. Puts 16K of software on line at ALL TIMES! Kit features a top quality soldermasked and silk-screened PC board and first run parts and sockets. All parts (except 2708's) are included. Any number of EPROM locations may be disabled to avoid any memory conflicts. Fully buffered and has WAIT STATE capabilities.

OUR 450NS 2708's ARE $8.95 EA. WITH PURCHASE OF KIT

ASSEMBLED AND FULLY TESTED ADD $25

16K STATIC RAM KIT-S 100 BUS$  
$295 KIT

FULLY STATIC, AT DYNAMIC PRICES

WHY THE 2114 RAM CHIP?
We feel the 2114 will be the next industry standard RAM chip (like the 2120 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 Bus! 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, some among other 4K Static RAM's the 2114 stands out! Not all 4K static RAMs are created equal! Some of the older 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.

BLANK PC BOARD W/DATA---$33
LOW PROFILE SOCKET SET----$12
SUPPORT ICS & CAPS---$19.95
2114 RAM's---8 FOR $69.95

16K DYNAMIC RAM CHIP
16K X 1 Bits, 16 Pin Package. Same at MOS 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 new, fully buffered contact fingers mfg. VERY LIMITED STOCK!
8 FOR $89.95

450 NS! 2708 EPROMS
Now full speed! Prime new units from a major U.S. Mfg. 450 NS. Access time. 1K x 8. Equiv. to 4-1702 A's in one package.

$16.75 ea. 4 FOR $50.00
PRICE CUT

8K LOW POWER RAM KIT-S 100 BUS$  
$129 KIT

ADD $5 FOR 250NS!

(450 NS RAMS)

Thousands of computer systems rely on this rugged, work horse, RAM board. Designed for error-free, NO HASSLE, systems use.

KIT FEATURES:
1. Doubled sided PC Board with solder mask and silk screen layout. Gold plated contact fingers.
2. All sockets included.
3. Fully buffered on all address and data lines.
4. Phantom is jumper selectable to pin 87.
5. FOUR 7805 regulators are provided on card.

Blank PC Board w/Documentation $29.95
Low Profile Socket Set...13.50
Support ICS's (TTL & Regulators) $9.75
Bypass CAP's (Disc & Tantalums) $4.50
ASSEMBLED AND FULLY BURNED IN ADD $30

TM990 BUSS PROTOTYPE & WIREWRAP BOARD
For use with the Texas Instrument Series of 16 Bit Microcomputer Module. Fully bus compatible. An inexpensive and quick way to expand the capacity of your TI computer. Made of G-10 Epoxy PC material. Gold plated contact finger, all plated through holes. High density, up to 100 DIP's. Fully documented.

$70 each (OEM Discounts Available)

NATIONAL SEMICONDUCTOR
CAR CLOCK MODULE - #MA6008
$6.99 each

NEW!

Originally used by HYGAIN to indicate time and channel on an expensive C.B. Mini size, self contained module. Not a Kit. Four digits plus flashing indicator for seconds. Includes MM5369 and 3.58 MHz crystal for super accurate time base. With hookup data.

INCLUDES CRYSTAL/ TIMEBASE!
WORKS ON 12 VDC!

$75 ea. 3 FOR $22 KLM-1

GENERAL INSTRUMENT  
FULL WAVE BRIDGE  
4 AMP 600 PIV
3/4 IN. SQUARE - WITH LED

EXPERIMENTER'S HEATING PLATE
Large Mfgs. Surplus. 5x10½ in. Made of 3/8 in. tempered glass with heating element laminated on back. Works off 120 VAC. Protected by thermostat and two thermal fuses. Rated 120 Watts. Use for any heating applications. Perfect for heating ferric chloride to increase PC Board etching efficiency. Units are brand new, non-submersible.

WHILE THEY LAST--$2.99 each

MALLORY COMPUTER GRADE CAPACITOR
30,000 MFD 15 WDVC
Small: 3 x 2 Inches
$1.99 ea. 3 FOR $4.99
NEW!

ELECTRONIC COMPUTER CLOCK CHIPS
N.S. M5S313. Features RC7 7 segment and BCD outputs. 28 Pin DIP.
$4.95 with Data

FAIRCHILD SUPER JUMBO LED READOUT
A full 80 inch character. The biggest readout we have ever sold! Super efficient. Compare at up to $2.95 each from others. YOUR CHOICE

THE COLOSSUS
FULL 80 CHARACTERS

CIRCLE 100 ON INQUIRY CARD

PAGE 134"
The DATA-TRANS 1000

A completely refurbished IBM Selectric Terminal with built-in ASCII Interface.

$1395

**Features:**
- 300 Baud
- 14.9 characters per second printout
- Reliable heavy duty Selectric mechanism
- RS-232C Interface
- Documentation included
- 60 day warranty - parts and labor
- High quality Selectric printing
- Off-line use as typewriter
- Optional tractor feed available
- 15 inch carriage width

**HOW TO ORDER**

**DATA-TRANS 1000**

1. We accept Visa, Master Charge. Make cashiers checks or personal check payable to:

**DATA-TRANS**

2. All orders are shipped F.O.B. San Jose, CA

3. Deliveries are immediate

**TIDMA**

- Tape Interface Direct Memory Access
- Converts video to AM modulated RF, Channels 2 or 3. So powerful almost no tuning is required. On board regulated power supply makes this extremely stable. Rated very highly in Doctor Dobbs' Journal. Recommended by Apple. Power required is 12 volts AC C.T., or +5 volts DC. Board only $7.80 part No. 107, with parts $13.50 Part No. 107A

**T.V. INTERFACE**

- Converts video to AM modulated RF, Channels 2 or 3. So powerful almost no tuning is required. On board regulated power supply makes this extremely stable. Rated very highly in Doctor Dobbs' Journal. Recommended by Apple. Power required is 12 volts AC C.T., or +5 volts DC. Board only $7.80 part No. 107, with parts $13.50 Part No. 107A

**ASCII KEYBOARD**

Mounted to DECWRITER Panel

By Cherry Products ... ASSEMBLED $9.95

This bidirectional board is a direct replacement for the board inside the Trendata 1000 terminal. The on board connector provides RS-232 serial in and out. Sold only as an assembled and tested unit for $330.00. Part No. TA 1000C

**ASCII to Correspondence code converter**

This bidirectional board is a direct replacement for the board inside the Trendata 1000 terminal. The on board connector provides RS-232 serial in and out. Sold only as an assembled and tested unit for $330.00. Part No. TA 1000C

**To Order**

- Mention part number, description, and price. In USA, shipping paid for orders accompanied by check, money order, or Master Charge. BankAmericard, or VISA number, expiration date and signature. Shipping charges added to C.O.D. orders. California residents add 6.5% for tax. Outside USA add 10% for air mail postage and handling, no C.O.D.'s. Checks and money orders must be payable in US dollars. Parts kits include sockets for all ICs, components, and circuit board. Documentation is included with all products. Prices are in US dollars. No open accounts. To eliminate tariff in Canada boxes are marked "Computer Parts." Dealer inquiries invited. 24 Hour Order Line: (408) 263-9246

**ELECTRONIC SYSTEMS**

Dept. B, P. O. Box 21638, San Jose, CA USA 95151

Circle 125 on inquiry card.

Circle 126 on inquiry card.
**MONODEM**
- Type 103
- Full or half duplex
- Works up to 300 baud
- Originate or Answer
- No coax, only low cost components
- TTL input and output
- Connects 8 line speaker and crystal micro directly to board
- Uses XRS DTMF demodulator
- Requires +5, -12 VDC
- Board only $19.95

**APPLE III**
- SERIAL I/O INTERFACE
- Baud rate is continuously adjustable from 0 to 3000
- Plug into any peripheral connector
- Low current drain
- RS-232 input and output
- Connects 8 line speaker and crystal micro directly to board
- Uses XRS DTMF demodulator
- Requires +5 volt supply
- All connections provided, see below.

**8K EPROM PICPEON**
- Saves programs on PROM permanently
- Until erased via UV light
- Up to 8K bytes
- Programs may be directly run from the program saver
- Such as fixed routines or assemblers
- S-100 bus compatible
- Room for 8K bytes of EPROM non-volatile memory
- On-board PROM programming
- Address relocation of each 4K of memory to any 4K boundary
- jumper selectable
- Switch selectable 5 volt or +5 volt option
- “Turkey” systems and computers
- Without a front panel
- Program saves software available
- Full silkscreen for easy assembly
- Board only $39.95

**S-100 BUS ACTIVE TERMINATOR**
- Board only $14.95
- Part No. 800, with parts $42.95

**WAMECO PRODUCTS**
- WITH ELECTRONIC SYSTEMS PARTS
- PB-1 FLOPPY BOARD CONTROLLER
  - Drive signals paper, paper, paper, paper, paper, paper, paper, paper, paper
  - Board only $35.00
  - Part No. 800, with parts $62.95

**DC POWER SUPPLY**
- Board supplies a regulated +5 volts at 3 amps, +12 volts at 3 volts
  - Power required is
  - 8 volts AC at 3 amps, and 24 volts AC CT, at 1.5 amps
  - Board only $12.50
  - Part No. 805, with parts $24.50

**NOTE:**
- Mention part number, description, and price. In USA, shipping paid for orders accompanied by check, money order, or Master Charge, Bank Americard, or VISA
- Number, expiration date and signature. Shipping charges added to C.O.D. orders. California residents add 8.5% tax. Outside USA add 10% for air mail postage and handling. 0.C.O.D. and money orders must be payable in US dollars. Parts lists include sockets for all IC's, components, and circuit board. Documentation is included with all products. Prices are in US dollars. No open accounts. To eliminate unit in Canada boxes are marked "Computer Parts." Dealer inquiries invited. 24 Hour Order Line: 408/286-5046

**For free catalog including parts lists and schematics, send a self-addressed stamped envelope.**

**ELECTRONIC SYSTEMS**

Dept. B, P. O. Box 21638, San Jose, CA USA 95151
FOR TRS-80™ OWNERS
T-PAL
Programming Amateur’s Letter
THE “DO-IT-YOURSELF” SOFTWARE NEWSLETTER
Published Monthly. We’ll Teach You All The Latest Wrinkles — How To Get The Most Out Of Your Computer
Graphics - Games - Personal Home - Business - Finance And Much More
$24/year ... or write for FREE Details

THE MAIL MART
Box B 11102 San Francisco, CA 94101

BYTES’ New Toll-free Subscriber
W.A.T.S. Line
(800) 258-5485
To further improve service to our customers we have installed a toll-free WATS line in our Peterborough, New Hampshire office. If you would like to order a subscription to BYTE, or if you have a question related to a BYTE subscription, you are invited to call (800) 258-5485 between 8:30 AM and 4:30 PM Eastern Time. This applies to calls from within the continental US only.
We thank you and look forward to serving you.

CROMEMCO
AT 15% DISCOUNT
- EVERYTHING ON THE CROMEMCO CATALOG
- ALL NEW EQUIPMENT FULLY GUARANTEED
- MANY SOFTWARE PACKAGES AVAILABLE

DIGITAL DATA SYSTEMS, INC.
1396 N W 65 TERR
PLANTATION, FLA 33313
(305) 792-3290

C G ENGINEERING
P.O. BOX 1145
MILPITAS, CA 95035

SMART CASSETTES

List $1.00 10 for $7.50 50 for $32.50

MICROSETTE CO.
777 Palomar Ave. - Sunnyvale, CA 94086

Duplication Services
Microsette also offers professional duplication services for Commodore PET and Radio Shack TRS-80 Level I and Level II cassettes. Our service provides mastering, quality control, all material including two-piece box, affixing of your labels or supplying our blank labels and shipping. Prices start at $2.00 each in 100 quantity.

MICROSETTE CO.
777 Palomar Ave. - Sunnyvale, CA 94086

Professional
Real Estate Programs
available now on cassette
for Apple & TRS-80 II
- Property Management System (Two 8K Programs)
  1.) Expense Analysis
  2.) Income Info & Tracking
Introductory Offer '50
Price After July 15, '75
- Home Purchase Analysis
- Income Property Cashflow/Leverage
- Construction Cost/Profit
- Tax Free Exchange Model
Separately '20 All '65

Call (213) 372-8419 for immediate C.O.D.
Atrtractive Dealer Pricing

TRIPLE E TIME MACHINE
An IEEE 488 to $100 Bus Interface and Control Board. Included are counter-timer functions. Board is addressable anywhere in memory or I/O.
KIT $225 BARE $35

GEMINI SBC 3.0
A 2650 Microprocessor based single board computer for the $100 Bus. Tape interface, parallel input port, power control port, 3 to 7K ROM and 1 to 3K RAM on board. Documentation with BARE board.
A KIT (IK/IK) $275 BARE $35
B KIT $225

MONITOR IX machine language monitor in 2708. Supported are: memory inspect and change; register inspect; breakpoints; tape load and dump; memory map; video and keyboard input port.
$20

C C ENGINEERING
P.O. BOX 1145
MILPITAS, CA 95035

Circle 323 on inquiry card.

Circle 229 on inquiry card.

Circle 199 on inquiry card.

Circle 136 on inquiry card.

Circle 310 on inquiry card.

Circle 105 on inquiry card.

Circle 74 on inquiry card.
PERCOM SAMPLER

For your SS-50 bus computer — the CIS-30+
• Interface to data terminal and two cassette recorders with a unit only 1/10 the size of SWTP's AC-30.
• Select 30, 60, or 120 bytes per second cassette interfacing, 300, 600 or 1200 baud data terminal interfacing.
• Optional mod kits make CIS-30+ work with any microcomputer. (For MITS 680b, ask for Tech Memo TM-CIS-30—09.)
• KC-Standard/Bi-Phase-M (double frequency) cassette data encoding. Dependable self-clocking operation.
• Ordinary functions may be accomplished with 6800 Mikbug™ monitor.
• Prices: Kit, $79.95; Assembled, $99.95.

In the Product Development Queue...

Coming PDQ. Watch for announcements.
6809 Processor Card — With this SS-50 bus PC board, you'll be able to upgrade with the microprocessor that Motorola designers describe as the "best 8-bit machine so far made by humans."
The Electric Crayon™ — This color graphics system includes its own µP and interfaces to virtually any microcomputer with a parallel I/O port.
Printer Interface — For your TRS-80™. Interface any serial RS232 printer to your TRS-80™ with this system.

For your S-100 computer — the CI-812
• Both cassette and data terminal interfacing on one S-100 bus PC board.
• Interfaces two recorders. Record and playback circuits are independent.
• Select 30, 60, 120, or 240 bytes per second cassette interfacing, 110 to 9600 baud data terminal interfacing.
• KC-Standard/Bi-Phase-M (double frequency) encoded cassette data. Dependable self-clocking operation.
• Optional firmware (2708 EPROM) Operating System available.
• Prices: kit, $99.95; assembled, $129.95.

CASSETTE SOFTWARE
For 8080/80-µCs...
BASIC ETC — Developed by the co-authors of the original Tiny BASIC, BASIC ETC is easy to use yet includes commands and functions required for powerful business and scientific programs as well as for hobby applications. cassette compatible for re-editing. Prints or displays full assembly-type output listing.
• Prices include a comprehensive instruction manual. Also available: Test Cassette, Remote Control Kit (for program control of recorders), and an IC Socket Kit are also available.

If you're programming on a 6800 µC, you'll want these development and debugging programs written by Ed Smith of the Software Works.
Disassembler/Source Generator — Disassembles SWTP Record Assembler, TSC Mnemonic Assembler/Text Editor or Smoke Signal Mnemonic Assembler/Text Editor and produces compacted source code suitable for re-editing. Prints or displays full assembly-type output listing.
• Prices include a comprehensive instruction manual. Also available: Test Cassette, Remote Control Kit (for program control of recorders), and an IC Socket Kit are also available.

For your data storage — Pilon-30™ and Pilon-10™ data cassettes
• Orders-of-magnitude improvement in data integrity over ordinary audio cassettes.
• Pilon-coated pressure pad eliminates lint-producing felt pad of standard audio cassettes.
• Smooth pilon coating minimizes erratic tape motion.
• Foam pad spring is energy absorbing. Superior to leaf spring mounted pad which tends to oscillate and cause flutter.
• Five-screw case design virtually precludes deformation during assembly.
• Price: $2.49.

SS-50 Prototype Cards:
large card (up to 70 40-pin ICs) $24.95
1/0 size card ....... $14.95

To order products or request additional literature, call Percom's toll-free number: 1-800-327-1592. For technical information call (214) 272-3421.
The EXPANDORAM is available in versions from 16K up to 64K, so for a minimum investment you can have a memory system that will grow with your needs. This is a dynamic memory with the invisible on-board refresh, and it WORKS!

- Bank Selectable
- Phantom
- Power 8VDC, ±16VDC, 5 Watts
- Lowest Cost Per Bit
- Uses Popular 4116 RAMS
- PC Board is doubled solder masked and has silk-screen parts layout.

**SD EXPANDORAM**

The Ultimate S-100 Memory

- Extensive documentation clearly written
- Complete Kit includes all sockets for 64K
- Memory access time: 375ns, Cycle time: 500ns.
- No wait states required.
- 16K boundaries and Protection via Dip Switches.
- Designed to work with Z-80, 8080, 8085 CPUs.

**EXPANDO 64 KIT (4116)**

- 16K $245.00
- 32K $310.00
- 48K $375.00
- 64K $440.00

**SAVE $100**

**DM2700S DISK & CABINET with POWER SUPPLY**

DM2700S includes Siemens FD120-8" Disk Drive with the following features:

- Single or Double Density
- Hard or Soft Sector
- Door Interlock
- Write Protect
- Hard Sector Detection
- Data Cable
- 500 KBS Transfer
- Drive of FM
- Bit density 6536 BP1
- Sugar 800 Series Compatible

**DM2700S Disk Drive & Cabinet REG. $750 SALE PRICED**

$650.00

**DM2700 Cabinet, less Drive**

$225.00

**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, or CMOS, and Probe to test mode. Test logic levels and pulses through digital circuits. Even sketch and latch for pulse detection. Instant recognition of high, low or invalid levels, open circuits, and shorts. Model LP-1 comes complete with 11/2" dia. x 9" long probes, 12V to 15V square wave power supply, and LED, dual speed front panel pulse detector.

**MODEL LP-1**

- Hand-held logic probe provides instant reading of logic levels for TTL, DTI, CMOS, and other popular circuits.
- Tests with single pulse or 100 pulses per second.
- Built-in LED pulse detector (LED): High Speed Logic Probe. Captures pulses as short as 10 ns Input Impedance: 100,000 ohms.
- Maximum Input Signal (Frequency): 10 MHz. Pulse Memory: Pulse or level transition detected and stored.
- Pulse Memory: Pulse or level transition detected and stored.
- Pulse Memory: Pulse or level transition detected and stored.

**CASES**

**DEC LSI-11, POPS, POP11, Heat H-11, P Pattern Epox Glass, Plug Board 84.5, etc.**

- Dual 8-pin DECOG/CA.
- Connectors 8-14, 5-9, 24, 10, 22, 17, 19
- 1-2, 9-13, 46, 45, 42, 10, 12, 14, 15, 17
- Vector-PAK assembled microcomputer cases

**PRIORITY ONE ELECTRONICS**

16723B Roscoe Blvd. Sepulveda, CA 91343

Terms: Visa, MC, BAC, Check, Money Order, C.O.D. U.S. Funds Only. California residents add 6% sales tax. Minimum order $10.00. Prepaid U.S. orders less than $75.00 include 5% shipping and handling, minimum $2.50. Excess refunded. Just in case... please include your phone number.

Prices subject to change without notice.

We will do our best to maintain prices thru July 1979.

Phone orders welcome (213) 894-8171, (800) 423-5633

OEM and Institutional inquiries invited.

Visit our new retail location!
HICKOK LX303 $74.95

Dual-Trace 30 MHz Oscilloscope
Built in delay line for leading edge viewing of fast risetime pulses. Compact portable size. 11.7 ns risetime. Full time X4 expansion (horizontal position control automatically allows moving trace 4 screens within a 4 screen display). Test Most Digital Logic Circuits including Microprocessors. Stable triggering up to 50 MHz. High and low pass trigger filters. 11 step precision vertical attenuators. 24 calibrated sweep times. Analyze CM AB and SSB waveforms. Regulated power supplies for accuracy over 105-130 VAC

SPECIFICATIONS
Vertical Ranges: 10 mV/DIV to 200 mV/DIV in 11 calibrated steps. Variable control permits fine adjustment between steps. Accuracy +4%. Frequency Responses: DC to 30 MHz (3 dB) DC coupled. 2 Hz to 30 MHz (3 dB) AC coupled. Risetime: 11.7 ns. Overshoot; 4% or less. Positioning: 3 screens. Input Impedance: 1 megohm ± 2% shunted by 27 pF ± 1.0 pF. Maximum Input Voltage: 500 V DC plus peak AC except 300 volts on 0.1 V range. Vertical Modes: AC, DC only. Alternate A & B. Chopped A & B. Difference (A-B). Time Base: All calibrated steps. 2 SEC/DIV to 0.5 SEC/DIV in 24 calibrated steps. Variable control permits fine adjustment between steps. Accuracy: +4%. Except 7% from 2 SEC/DIV to 0.5 SEC/DIV. TRIGGERING MODES: AC-CH, High pass filter, signal component below 5 kHz rejected. Auto: Provides continuous sweep without input signal. Sources: Line, Internal, External. Slope: Positive and negative; continuously variable triggering control. Sensitivity: Internal 0.25 V/DIV; 1 division to 50 MHz; External 200 MV to 5 V peak-to-peak. EXTERNAL HORIZONTAL (X-Axis): Frequency Response: DC to 5 MHz, DC coupled. Impedance approximately 30 pF. GENERAL: Probe Calibrator: 0.6 V peak-to-peak. 200 ns risetime. CRT: 4-inch flat face monitor with viewing area of 8 x 10 divisions. Power Requirements: 105-125 V, 50-400 Hz, 35 watts. SIZE & WT.: 6 7/8" h x 11 1/4" w x 17 1/2" d. 27 pounds (not including handle). (17.2 cm x 28.6 cm x 44.2 cm) 17.2 kg) ACCESSORIES: Model 532 includes 2 model SP7, 10.1 probes and Instruction manual. Includes two probes.

HICKOK 1c Sale

DUAL-TRACE 30 MHz Oscilloscope
Model 532 Deluxe 10:1 Probe Set
SALE PRICED AT $11450

price:$29.50

1/16" Vector BOARD .042 dia holes on .1 spacing for IC's

price:$18.86

PANAVISE 1144LHPC
4096 - BIT (1024x4) 450 ns Low Power Static Ram
8 for $50.00

price:$1.34

ORDER TOLL FREE
1-800-423-5633
(except CA, AK, HI, Call (213) 894-8171)

HICKOK LX303 $74.95
### Sockets Purchased in Multiples of 50

Sockets may be purchased in multiples of 50 per type for

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.4</td>
<td>6.75</td>
<td>50 Pin Male</td>
</tr>
</tbody>
</table>

### Contact Center Connectors

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.5</td>
<td>9.90</td>
<td>50 Pin Male</td>
</tr>
</tbody>
</table>

### Trend Settings Dividers

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.6</td>
<td>14.90</td>
<td>50 Pin Male</td>
</tr>
</tbody>
</table>

### Available with standard IC sockets.

### Fully assembled and tested.

### Integral molded-on strain relief.

### Line-by-line probability.

### P A D Jumpers are the low-cost, high-quality solution for jumpering within a PC board.

### Interconnecting between PC boards, backplanes and motherboards, interfacing input and output signals, and more.

### All assemblies use rainbow cable. Standard lengths are 6, 12, 18, 24 and 36 inches.

### 3 LEVEL GOLD WIRE WRAP SOCKETS

Sockets purchased in multiples of 50 per type may be combined for best price.

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>1.9</td>
<td>13.45</td>
<td>50 Pin Male</td>
</tr>
</tbody>
</table>

### GOLD SOLDE TAIL STANDARD IC SOCKETS

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>2.1</td>
<td>19.90</td>
<td>50 Pin Male</td>
</tr>
</tbody>
</table>

### TIN SOLDER TAIL - LOW PROFILE IC SOCKETS

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Male</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Female</td>
</tr>
<tr>
<td>3.4</td>
<td>28.90</td>
<td>50 Pin Male</td>
</tr>
</tbody>
</table>

### 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 probability.

### A P A D Jumpers are the low-cost, high-quality solution for jumpering within a PC board.

### Interconnecting between PC boards, backplanes and motherboards, interfacing input and output signals, and more.

### All assemblies use rainbow cable. Standard lengths are 6, 12, 18, 24 and 36 inches.

### PRIORITY ONE ELECTRONICS

**16723 Soyroc 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 $100.00. Prepaid U.S. orders less than $75.00 include 5% shipping and handling, minimum $2.50. Excess refunded. Just in case... please include your phone no.

**Prices subject to change without notice.**

**GEM and Institutional**

We will do our best to maintain prices thru July 1979. Inquiries invited.

**Phone orders welcome (213) 894-8171, (800) 423-5633**

**Circle 312 on inquiry card.**
CA LIFORNIA COMPUTER SYSTEMS

16K RAM BOARD. Fully buffered addressable in 4K blocks. IEEE standard for bank addressing 2114s.
PCBD ........................................................ $26.95

Kit 450 NS/EC .................................................................$29.95

PT-1 FLOTO BOARD. Over 2,600 holes 4” regulators. All S-100 bus functions labeled, gold fingers.
PCBD ........................................................ $29.95

PT-2 FLOTO BOARD. Similar to PT-1 except set up to handle solder tail pockets.
PCBD ........................................................ $29.95

FOREVER CYBERنك/SOLID STATE MUSIC.

PB-1 2706 & 2716 Programming Board with provisions for 4K or 8K EPROM. No external supplies required. Text sockets. Kit .... $124.95

CB-1 8060 Processor Board. 2K of PROM 256 BYTE RAM power on rest Vector Jump Parallel port with status Kit .... $119.90 PCBD .... $30.95

MB-6B Basic 8K8 ram uses 2102 type rams, S-100 bus. Kit 450 NS/EC .... $139.95 PCBD .... $25.95

MB-7 16K8, Static RAM uses 254 RAM, fully buffered Kit ........................................................................ $259.95

MB-8A 2706 EROM Board, S-100, 8K8 or 16Kx8 kit without PROMS $75.00 PCBD $29.95

MB-9 4K8 RAM/PROM Board uses 2112 RAM or 82129 PROM kit without RAMS or PROMS $72.00

IO-2 S-100 8 bit parallel /10 port, 5/6 of boards is for kludging. Kit .... $46.00 PCBD .... $29.95

IO-4 Two serial I/O ports with both handshaking 20/50 ma current loop: Two parallel I/O ports. Kit .... $130.00 PCBD .... $23.95

VB-1B 64 x 16 video board, upper case Greek, composite and parallel video with software, S-100. Kit .... $125.00 PCBD .... $29.95

Altair Compatible Mother Board, 11 x 11 x 1¼”. Board only .... $39.95. With 15 connectors .... $94.95

Extended Board full size. Board only .... $9.49

With connector .... $13.45

SP-2 Synthesizer Board S-100 PCBD ........................................ $42.95 KIT ................................................. $135.95

WITH WAMECO AND CYBERCOM PCBDS

PREMIER PRODUCTS: 80 CHARACTER VIDEO BOARD, 10 BOARD WITH CASSETTE INTERFACE.

DEALER INQUIRIES INVITED, UNIVERSITY DISCOUNTS AVAILABLE AT YOUR LOCAL DEALER
**MICROBYTE 280/1-O**
- A complete single board Z80A CPU and serial parallel I/O system
- IBM 801 S-100 Bus compatible, IMSAI, ALTAIR
- 280A CPU (IMS version of the Z80)
- 150 instructions — superior of and upward compatible from the 8080's 76 instructions
- 1K/2K/4K 4K PROM onboard with 280A Monitor program
- On board EPROM can be hard-
  ware and/or software deselected
- 2MHz or 4MHz operation is
  switch selectable.
- 0 or 1 wait state for all cycles is
  switch selectable.
- Full vectored interrupt capability
  (1 interrupt with NMI (1 bit)
- 2 RS-232C serial ports with 2821
  USART
- Serial baud rates switch selectable
- 1 8-bit parallel port with 2 control
  lines with 280A-PID

**8251**
- PROGRAMMABLE/U-ART
- TESTED @ 4 MHZ
- $6.00 each

**IMSAI CONN.**
- 100 PIN-SOLDERTAIL
- GOLD CONTACTS
- $3.25 each or 10 / 2.75 ea.

**SA400**
- DISK DRIVE INCLUDES
- CABINET, NO PWR SUPPLY
- CUTOUTS FOR SWITCH, FUSE, & INTERFACE CABLE
- Ltd. Qty.

**MICROBYTE 16K STATIC RAM BOARD**
- Fully S100 Bus Compatible, IMSAI, SOL, ALTAIR, ALPHA MICRO
- Uses National’s Low Power 5257
  4Kx1 Static Rams
- 2 MHz or 4 MHz operation
- On board single 5 amp regulator
- Thermally designed heat sink
  (board operating temperature
  0°-70°C)
- Inputs fully low power Shottky
  Schmitt Trigger buffered on all
  address and data lines
- Phantom jumper selectable to
  pin 67
- Each 4K bank addressable to any
  4K slot with in a 64K boundary.
- 4K hardware or software selec-
  table
- Selectable port address
- 4K banks can be selected or
disable on power on clear or reset

**REGULATORS**
- 1-9 10-49 50 up
- 320 T-5 .90 .85 .75
- 320 T-9 .90 .85 .75
- 340 T-12 .85 .80 .70
- 78 H05 6.50 5.85 5.50

**SOCKETS—LO PROFILE**
- (in)
- 1-24 25-99 100-499 500 up
- 14 PIN .18 .17 .15 .13
- 16 PIN .19 .18 .17 .14
- 18 PIN .24 .23 .20 .18
- 20 PIN .29 .28 .26 .25
- 24 PIN .34 .33 .32 .30
- 40 PIN .60 .58 .56

**MICROBYTE 32K STATIC RAM BOARD**
- Fully S100 Bus Compatible, IMSAI, SOL, ALTAIR, ALPHA MICRO
- Uses National’s Low Power 5257
  4Kx1 Static Rams
- 2 MHz or 4 MHz operation
- On board single 5 amp regulator
- Thermally designed heat sink
  (board operating temperature
  0°-70°C)
- Inputs fully low power Shottky
  Schmitt Trigger buffered on all
  address and data lines
- Phantom jumper selectable to
  pin 67
- Each 4K bank addressable to any
  4K slot with in a 64K boundary.
- 4K hardware or software selec-
  table
- One on board 8-bit output port
  enables or disables the 32K in 4K
  blocks
- Selectable port address
- 4K banks can be selected or
disable on power on clear or reset

**CALL FOR INFO**
- ON OUR NEW
**PROGRAMMER**
- PROGRAMS 2708 / 2716 / 2732

**SHUGART**
- 801-Disk Drive
- WITH CABINET & POWER SUPPLY
- ASSEMBLED & TESTED
- 1 YR PARTS & LABOR
- Mfg. By Lobo Drive
- $585.00

**ORDERING INFORMATION:**
- Name, Address, Phone
- Ship by: UPS or P.P.
- Shipping Charge: Add $2.50 up to
  5 lbs., all excess shipping charges
  will be refunded. Credit cards will be
  charged appropriate freight.

**TERMS:**
- We accept cash, check, money
  orders, Visa, and Master Charge
  cards.
- COD's: on approval only
- Open Aceit's: companies may
  inquire for net terms.
- Tax: add 6% for Calif. residents only

**SHUGART INTERFACE ASSEMBLED (board operating temperature
- 0°-70°C) & TESTED WITH 1 YR. WARRANTY• Inputs fully low power Shottky
- (tin)
- 1.24 2.30 4.50 5.60 7.65
- 10-49 50 up
- 320 100-499
- 14 PIN .18 .17 .15 .13
- 16 PIN .19 .18 .17 .14
- 18 PIN .24 .23 .20 .18
- 20 PIN .29 .28 .26 .25
- 24 PIN .34 .33 .32 .30
- 40 PIN .60 .58 .56

**SHUGART CALL FOR INFO**
**MICROBYTE DISK CONTROLLER**
- IBM 3740 Soft Secteded
- Compat.
- or 8080 compatible on S-100
- Bus
- Single density runs both mini
  and full size drives, runs CPM, on
  Shugart, Persci., Memorex etc.

**MICROBYTE MOTHERBOARD**
- Active Diode termination
- Slot for IMSAI front panel
- Terminal block connection for easyhook-up
- 9 SLOT $135.00
  20 SLOT $200.00

**SHUGART CALL FOR INFO**
**1 KNOTT STREET**
**GARDEN GROVE, CALIFORNIA 92641**
**(714) 891-2663**
Rockwell AIM-85: The Head-Start in Microcomputers

A KIM-1 compatible machine with on-board printer and a real keyboard!

- $375.00 w/1K RAM
- $450.00 w/4K RAM

4K assembler/editor in ROM: $80.00
8K BASIC in ROM: $100.00
Power supply: $59.95
Case for AIM-85: $49.95

Special Package Price: $599.00
AIM-65 (4K), Power Supply, Case, and 8K BASIC ROM

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE PIGGY MAINFRAME

THE PIGGY IS HERE!

This sleek new mainframe is neatly trimmed to hold six S-100 boards, two mini-floppy drives, and is available in five colors. Power requirements: 115/220 VAC, 50/60 Hz. Weight: 27 lbs. (with drives). Dimensions: 21.375" Wide x 8.4" High x 15.875" Deep. Power Supply: 8 volts ± 18 amps unregulated, +16 volts ± 3 amps regulated, -16 volts ± 3 amps regulated, +5 volts ± 3 amps regulated, +12 volts ± 3 amps regulated.

THE PIGGY (Without Drives) ..... $475.00

LEEXD MONITOR

- 12" Black and White
- 12 MHZ Bandwidth
- Handsome Plastic Case
- $139.00

VECTOR PLUG BOARDS


5.3" x 10" x 1 1/16" ..... $19.95

Same as above, except Patter, Less Power Buses & Heat Sink. ..... $15.95

Part No. positions 1-9
SWD-103 0 $1.18
SWD-104 1 $1.30
SWD-105 5 $1.24
SWD-106 6 $1.28
SWD-107 7 $1.30
SWD-108 8 $1.34
SWD-109 9 $1.36
SWD-110 10 $1.38

Hi-Density Dual-In-Line Plugboard For Wire Wrap With Power & Ground Bus. Epoxy Glass 1/16" 44 Pin Con Spaced .156

3682-2-9.5 x 4.5... $10.97
3682-2-6.5 x 4.5... $9.81

Gen Purpose D.I.P. Boards With Bus Pattern For Solder Or Wire Wrap. Epoxy Glass 1/16" 44 Pin Conn. Space .156

3677-9.5 x 4.5... $10.00
3677-6.5 x 4.5... $9.74

CARD EXTENDER

Card Extender has 100 Contacts 50 Per Side On 125 centers. Attached Connector is Compatible With S-100 Bus Systems

$52.83

SD SYSTEMS

Z-80 STARTER KIT

Based on the powerful 280 CPU, this kit is an ideal introduction to microprocessors. It has an on-board keyboard and display, plus cassette interface and expansion provisions for two S-100 connectors. This "Do-It-All" Board will also program the 271K EPROM.

Kit: $239.95
Assembled and Tested: $399.95

JAILE PROTO BOARD

Includes gold plated fingers, S-100 Sizing, Expandable To 64K 4116 (8Kx2, 250ns) Dynamic RAM Boards

Kit: $159.95
Assembled and Tested: $299.95

FLOPPY DISK INTERFACE

JADE Floppy Disk (Tambell board)

Kit: $195.00
Assembled & Tested: $250.00

FLOPPY DISK INTERFACE

JADE-100 Plugboard

Kit: $159.95
Assembled and Tested: $299.95

JADE PARALLEL/SERIAL INTERFACE

S-100 compatible 2 Serial I/O ports, Parallel I/O, Kit JG-P/S: $124.95
Assembled & Tested: $179.95
Bare Board w/Manual: $30.00

SOLID STATE MUSIC'S I/O 4

2-Serial & 2-Parallel I/O Ports, S-100 with full hand-shaking.

KIT: $195.95
Assembled: $199.95
Bare Board: $29.95

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.

With 16K of memory ..... $1150.00

THE EXIDY SORCERER

Flexibility is the key. The Sorcerer Computer gives you the flexibility of using ready-to-run, pre-packaged programs or doing your own thing and personalizing the programs for yourself. Whichever you choose, the Sorcerer is the personal computer that speaks your language.

The Sorcerer also provides full graphics capabilities. Each character, formed by an 8 x 8 dot cell, can be programmed as a graphic symbol set. An analog to digital converter (512 x 256 addressable points) gives a total of 122,880 locations for super animation and extremely tight plotting curves. The alphanumeric set gives 44 x 30 characters on the video screen.
THE BIG Z
THE NEW Z-80
CPU BOARD FROM JADE

Features Include: • S-100 Compatible, available in 2MHz or 4MHz versions. • On-board 2708, 2716, 2516, or 2532 EPROM can be addressed on any 1K, 2K, or 4K boundary, with power-on jump to EPROM. • On-board EPROM may be used in SHADOW mode, allowing full 64K RAM to be used. • Automatic MWRITE generation if front panel is not used. • On-board USART for synchronous or asynchronous RS232 communication (on-board baud rate generator). • Reverse-channel capability on USART allows use with buffered peripherals or devices with "not-ready" signal.

2MHz: KIT: CPU-30200K, 2 lbs $149.95
Assembled and Tested: CPU-30200A, 2 lbs $199.95

4MHz: KIT: CPU-30201K, 2 lbs $199.95
Assembled and Tested: CPU-30201A, 2 lbs $209.95

MICROPROCESSORS

- F8 $16.95
- Z80 (2MHz) $10.95
- Z80A (4MHz) $14.95
- COP18602CD $9.95
- 6502 $11.95
- 6800 $9.95
- 6802 $14.00
- 6808-1 $15.95
- 6835 $24.00
- 6805-8 $24.00
- 6830-1 $10.00
- 6805 $23.00
- TM5800T $49.95
- 8080A SUPPORT DEVICES $29.95
- 8085 $23.00
- TMS9900TL $49.95
- 8080A SUPPORT DEVICES $29.95

POWER SUPPLIES

- PSD-204A: For Single 8" Disk Drive $12.75
- PSD-2650: By Power-One or Alpha Power $13.75
- MM7640 $18.00

FLOPPY DISK CONTROLLERS

- 177886: $39.95

DISK DRIVES

- 851 5½" $295.00

KEYBOARD CHIPS

- AYS-2736 $12.75
- AYS-3650 $13.75
- MM7640 $18.00

JADE'S DOUBLE DENSITY

- Single or Double Density Recording
- Full Size or Mini Floppy
- CP/M Compatible in either density
- Programmed Data Transfer, no DMA
- Controls up to 8 drives
- IBM format in either density
- Software Selectable Density

KIT: $249.00
Assemb. & Tst: $299.00

JADE'S CABLE ASSEMBLIES

- Mini-Disk Cable Kit: 5½"/1¼" interface to 2 Shugart or compatible drives. Cable is 5" long with 34 pin edge connectors WCA-3421K $34.95
- 8" Disk Cable Kit No. 1: 34 pin edge connector WCA-5031K $38.45
- 8" Disk Cable Kit No. 2: 50 pin edge connector (great for Tarbell disk controller) WCA-5032K $38.45
- Signal Cables: 6 feet long, 34 pin edge connectors at each end WCA-3421A $24.95

JADE'S COMPUTER PRODUCTS

- 4901 W ROSECRANS AVENUE
- HAWTHORNE CALIFORNIA 90250
- Telephone: (213) 679-3313
- (800) 421-5809 Continental U.S.
- (800) 262-1710 Inside California

Jade Special Package Deal $1225.00

Vista V60 Mini Disk SYSTEM
FOR TRS-80 $395.00

- Single density 5½, 35-Track drive. Cabinet and power supply available

- Micro Peripherals, Inc. Operates in either, single density (125KB, unformatted) or double density (250KB, unformatted) modes, up to 40 tracks, with a track-to-track access time of only 5 ms

- Controls up to 8 drives
- IBM format in either density
- Software Selectable Density

- KIT: $249.00
- Assem. & Tst: $299.00

- Two Drives Siemens/GSI 8" Floppy
- Power Supply for Above
- Jade Double Density Board (KIT)
- CP/M Operating System with Basic E
- Package of 10 Blank 8" Diskettes (Double Density)
- Includes Interface Cables

- Price if Purchased Separately $1544.95
- Jade Special Package Deal $1225.00

Circle 195 on inquiry card.
BUILD YOUR OWN LOW COST MICRO-COMPUTER POWER SUPPLIES FOR S-100 BUS, FLOPPY DISCS, ETC.

POWER TRANSFORMERS (WITH MOUNTING BRACKETS)

<table>
<thead>
<tr>
<th>ITEM</th>
<th>USED FOR</th>
<th>PRI. WINDING</th>
<th>TAPS</th>
<th>SECONDARY WINDING OUTPUTS</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO.</td>
<td>KIT NO.</td>
<td>2x8 Vac</td>
<td>2x14 Vac</td>
<td>2x24 Vac</td>
<td>SIZE</td>
</tr>
<tr>
<td>T1</td>
<td>1</td>
<td>0V, 110V, 120V</td>
<td>2x9A</td>
<td>2x2.5A</td>
<td>3x4.5A</td>
</tr>
<tr>
<td>T2</td>
<td>2</td>
<td>0V, 110V, 120V</td>
<td>2x12.5A</td>
<td>2x2.5A</td>
<td>3x4.5A</td>
</tr>
<tr>
<td>T3</td>
<td>3</td>
<td>0V, 110V, 120V</td>
<td>2x10A</td>
<td>2x2.5A</td>
<td>3x4.5A</td>
</tr>
<tr>
<td>T4</td>
<td>4</td>
<td>0V, 110V, 120V</td>
<td>2x4.5A</td>
<td>2x2.5A</td>
<td>3x4.5A</td>
</tr>
</tbody>
</table>

POWER SUPPLY KITS (OPEN FRAME WITH BASE PLATE, 3 HRS. ASSY. TIME)

<table>
<thead>
<tr>
<th>ITEM</th>
<th>USED FOR</th>
<th>+8 Vdc</th>
<th>-8 Vdc</th>
<th>+16 Vdc</th>
<th>-16 Vdc</th>
<th>+28 Vdc</th>
</tr>
</thead>
<tbody>
<tr>
<td>KIT 1</td>
<td>18 CARDS SOURCE</td>
<td>2A</td>
<td>2A</td>
<td>3A</td>
<td>3A</td>
<td>12x6x4&quot;</td>
</tr>
<tr>
<td>KIT 2</td>
<td>SYSTEM SOURCE</td>
<td>25A</td>
<td>2A</td>
<td>2A</td>
<td>2A</td>
<td>18x7x4&quot;</td>
</tr>
<tr>
<td>KIT 3</td>
<td>DISC SYSTEM</td>
<td>25A</td>
<td>2A</td>
<td>2A</td>
<td>2A</td>
<td>12x8x4&quot;</td>
</tr>
<tr>
<td>KIT 4</td>
<td>DISC SOURCE</td>
<td>25A</td>
<td>2A</td>
<td>2A</td>
<td>2A</td>
<td>12x8x4&quot;</td>
</tr>
</tbody>
</table>

SUNNY INTERNATIONAL (TRANSFORMERS MANUFACTURER)

STORE: 7245 E. ALONDRA BLVD. PARAMOUNT, CALIF. 90723
STORE HOURS: 9 AM - 6 PM

COMMERCIAL GRADE PERIPHERALS FOR THE MICRO-COMPUTER

PRINTER TERMINALS

- ASCII SELECTRIC PRINTER/TYPewriter: Why settle for less than letter-quality printout from your computer? Refurbished IBM Model 725 can be used as offline typewriter or on-line printer. Complete with solenoid power supply, case and ASCII interface card (TTI to CPU parallel port). Interface includes programmable ASCII translation table on EPROM with up to 8 tables for use with various type styles. Feedback signals on completion of each print cycle insure fastest printing speed. Can be used as off-line typewriter or on-line printer. Complete with solenoids, B tables for use with various type spheres, feedback signals on completion of letter-quality printout from your computer. Refurbished IBM Model 725 $750.00

- MODEMS

- POS 103/202 "MIX or MATCH" MODEM: BELL 103 and/or BELL 202 FREQUENCIES: Unique POS control design permits use in one housing of both Bell-compatible 103 (0 - 300 baud) and 202 (0 - 1200 baud) modern modules originally made by VADIC Corp. for a telephone company subsidiary. FEATURES: RS-232 serial interface, auto-answer, auto-dial, LED display, telephone line interface via acoustic coupler, manual DAA, or auto-answer DAA (sold separately). FULLY ADJUSTED: no special tools required. 3,000 mile range over standard dial-up telephone lines.

- POS 103 MODEM (with Auto Answer, Auto Dial) ... $179.95
- POS 202 MODEM (Half-Duplex with Reverse Channel) ... $249.95
- POS 202 MODEM (Half-Duplex w/Rev. Ch., Auto-Answer) ... $279.95
- POS 103/202 MODEM (Auto-Answer, Auto-Dial) ... $399.95

- M-P0100 NRZI TAPE DRIVE CONTROLLER/FORMATTER: Designed as interface between S-100 bus and 9-track, 800 BPI, NRZI tape drive. Allows microcomputer to read and write IBM-compatible 3" magnetic tapes. Software provided for 8080 or 8085 systems. Requires modification for drives of various types. Price: (includes S-100 card, controller card, 10' cable, software listing) ... $750.00
- M-P0101 TAPE DRIVE by WILLARD LABS. 9-track, 800 BPI, NRZI format, 12"/sec., 1200 feet, (10 megabyte capacity) Fully tested and warranted $599.00
- CONV3RT 12" IBM OFFICE SELECTRIC TO DIP TYPEWRITER: Kit includes assembled solenoids, switches, relay, harness, motor driver PCB plus instructions for installation and mCPU interface ... $200.00

- DIGITAL CASSETTE DRIVE from GTE/IS Terminal: 1800 baud, 6'/sec., AC motor; fed/fed circuitry plus tape head, no read/write electronics $25
- FORMS TRACTORS, Moore Variable width "Form A-Liner" for printer terminals:
  a) Model 56SP for 15" carriage IBM Selectrics (new) ... $50.00
  b) Model 78 for IBM or DIABLO HY-DRAF or II printers (new) ... $90.00
  c) POWER SUPPLIES for Disk Drive, mCPU, tested under load shown:
    - No. 519 (w/fan & AC cord): $45 reg., $12V reg., $24V, $4A (10 lb.) ... $39.95
    - LAMBDIA No. LME56 w/OV protect: $65 reg., @ 3A (5 lb.) ... $69.95

COMMERCIAL GRADE PERIPHERALS FOR THE MICRO-COMPUTER

Call or write for details, quantity prices, catalog. 15 day return privilege PLUS 90 day no charge replacement of defective parts. All orders shipped from stock. No back orders, no substitutions. M/C & VISA accepted.
appII
$200 FREE ACCESSORIES
Buy a 48K Apple II. mention this ad and take $200 in
accessories free (if ordered together). This offer is
good for $150 on 32K and $100 on 16K Apple IIs. Now you
can enjoy all the best for less. We have just published our
brand new 48 page mini/micro catalog and its full of
great mini and micro computer buys.
The microcomputer section is packed with
illustrations and text presenting all that's new
in personal computing. All the items you see in
this ad are described in much greater detail.
Featured is the Apple II and the PET along with
the newest PET accessories and disk.
Other popular computers, computer peripherals,
supplies and books are described, along with
some special deals you can get only if you have this catalog.
To get your free catalog just write us at the
address listed at the bottom of this ad. Please
be sure to include the department number to
insure speedy delivery.

SPECIAL OF THE MONTH
FREE MINI/MICRO CATALOG
We've just published our brand new 48 page
catalog and its full of great mini and micro computer buys.

Apple II Accessories
Centronics Printer Interface ...................................... $225
Disk Controller .......................................................... $285
Second Disk Drive ..................................................... $285
Parallel Printer Card .................................................. $285
Communications Card ............................................... $285
Hi-Speed Serial Card ................................................... $285
Firmware Card ........................................................ $285
Hobby/Proto Card ...................................................... $285
Microvector RF Mod. ................................................... $285
Sanyo M5044 Recorder .............................................. $285

NCE/CompuMart, Inc.
1250 North Main Street, Department BY79
P.O. Box 8610 Ann Arbor, Michigan 48107

10 DAY RETURN PRIVILEGE
(313) 994-3200

CIRCLE 283 ON INQUIRY CARD.
FOR SALE: Digital Group system dual Ph-Decks, dress cabinets, Hitachi video display, DG keyboard. 34 K programmable memory, lots of software. Will ship $720. Mr. Michael D. Serder, 15475 W 40th Ave, Littleton CO 80127.

FOR SALE: Digital Group system 34 K programmable memory boards, dual Ph-Decks. Lots of software. Homemade mainframe cabinet with front panel. All documentation. This is a reliable system. Worth $1700. Asking $1100. Larv, 3069 Copenhagen, Riverside CA 92504. (714) 687 6725.

FOR SALE: TRS-80 Model 2B, $650. Complete with video monitor, power supply, keyboard, cas. Soft suit. Level 1 rack. 9 manuals, game software, all in perfect condition. I pay shipping. $875 Bruce Grenwald. 7056 Granada Dr, Flint MI 48504. (313) 733 1730 during business hours.

FOR SALE: IMSAI boards. 32 K programmable memory, $650. 16 K programmable memory, $450. Will sell each for $225. 4 K memory boards $50. Everything works. Asking price for all $195. Mr. John Mason, 2700 Balboa St, Austin TX 78757.

FOR SALE: Going to school, must sell. SWTPC CT-1024 video terminal. Cursor and serial boards. Ti keyboard, power supply, scrubbing. $35 GT 61 graphics. Power supply, MLP PRG jack, software, $150. PR 40 printer. New head. MLP, smoke $250. 4 K memory boards $50. Morrow's $100 Wunderbus mother board with ten edge connectors. $100. All in perfect working condition. If phone disconnected, please write. Chip Kroll, 1436 Tennyson, Houston TX 77005. (713) 667 6218.

FOR SALE: HP 67 programmable calculator, all stand addressable. Plus extra rechargeable battery pack and 100 blank program cards. Compare at $486 (plus tax). Everything goes in Modu-bag. UPS for $356 certified check or money order. Michael Craig, 105 E Edward St, Endicott NY 13760.

FOR SALE: Apple computer, 32 K, $750 or trade for new HP67 and $200. Send card to Rob Dinnell, 600 Park Ave 12A, Capitola CA 95010.


FOR SALE: 4 K programmable memory chips. 16 UPD4140 chips from converting my 8 K Exidy Sorcerer to 32 K. Make me an offer. Tom Krierer, 310 Pine Av, Canby MN 56220.

FOR SALE: MITS 4 K dynamic programmable memory assembled, tested, burned in. Also 2501 board (only one part implemented) assembled, tested, including MITS documentation. Best offers or trades (will accept partial). T Shew, 5F Nutmeg Knoll, Cokesbury MD 21030.

FOR SALE: Two Ithaca Audio 300 ns low power 8 K programmable memory boards (1060), $150 each. Electronic Systems television typewriter board, stand alone, assembled but never used, $130. Front money order or cashier's check takes any or all. If phone disconnected, please write. R Schuchman, 6221 Del Paso, San Diego CA 92120.

FOR SALE: Why buy a kit? Two fully assembled, 500 ns 8 K programmable memory boards (one SIMM 8 R, one Logos L) $125 apiece. Own IMSAI UCRI cassette board, $30. Henry Kapsenberg, 1175 W Baseline Rd, Claremont CA 91711.

FOR SALE: Heath compatible dot matrix printer 5-7, 80 or 96 columns, operate in serial or parallel modes from 110 to 1200 bps. Made by practical automation, this unit is a current production machine, not a discontinued or surplus. Plenty of documentation. It is similar to the IPS 745. Make of fer. R C Jackson, 201 Mortimer Ave, Rutherford NJ 07070. (201) 939 2421.

FOR SALE: Terminal computer portable terminal or direct connection terminals. Contain 80 column quiet printer, full ASCII keyboard, built-in modem and acoustic coupler, dual speed 110 and 300 bps, RS-332 and 20 mA and DAA interfaces, in a small suitcase size enclosure. Brand new condition with warranty. Will ship UPS. $8549.5 plus shipping. S Stoddard, 12 Kathy Dr, Popouzon CA 93662.

FOR SALE: Going to school, must sell. SWTPC CT-1024 video terminal. Cursor and serial boards. TI keyboard, power supply, scrubbing. $35 GT 61 graphics. Power supply, MLP, PRG jack, software, $150. PR 40 printer. New head. MLP, smoke $250. 4 K memory boards $50. Morrow's $100 Wunderbus mother board with ten edge connectors. $100. All in perfect working condition. If phone disconnected, please write. Chip Kroll, 1436 Tennyson, Houston TX 77005. (713) 667 6218.

FOR SALE: HP 67 programmable calculator, all stand addressable. Plus extra rechargeable battery pack and 100 blank program cards. Compare at $486 (plus tax). Everything goes in Modu-bag. UPS for $356 certified check or money order. Michael Craig, 105 E Edward St, Endicott NY 13760.

FOR SALE: Apple computer, 32 K, $750 or trade for new HP67 and $200. Send card to Rob Dinnell, 600 Park Ave 12A, Capitola CA 95010.

FOR SALE: MITS 4 K dynamic programmable memory assembled, tested, burned in. Also 2501 board (only one part implemented) assembled, tested, including MITS documentation. Best offers or trades (will accept partial). T Shew, 5F Nutmeg Knoll, Cokesbury MD 21030.

FOR SALE: Two Ithaca Audio 300 ns low power 8 K programmable memory boards (1060), $150 each. Electronic Systems television typewriter board, stand alone, assembled but never used, $130. Front money order or cashier's check takes any or all. If phone disconnected, please write. R Schuchman, 6221 Del Paso, San Diego CA 92120.

FOR SALE: Why buy a kit? Two fully assembled, 500 ns 8 K programmable memory boards (one SIMM 8 R, one Logos L) $125 apiece. Own IMSAI UCRI cassette board, $30. Henry Kapsenberg, 1175 W Baseline Rd, Claremont CA 91711.

FOR SALE: Heath compatible dot matrix printer 5-7, 80 or 96 columns, operate in serial or parallel modes from 110 to 1200 bps. Made by practical automation, this unit is a current production machine, not a discontinued or surplus. Plenty of documentation. It is similar to the IPS 745. Make offer. R C Jackson, 201 Mortimer Ave, Rutherford NJ 07070. (201) 939 2421.
### WIREWRAP

#### PRECUT WIRE

<table>
<thead>
<tr>
<th>#30 Wire Kits</th>
<th>#1</th>
<th>#2</th>
</tr>
</thead>
<tbody>
<tr>
<td>250 3&quot; 100 4½&quot;</td>
<td>$7.95</td>
<td>$19.95</td>
</tr>
<tr>
<td>250 3½&quot; 100 5&quot;</td>
<td>250 2½&quot; 250 5&quot;</td>
<td></td>
</tr>
<tr>
<td>100 4&quot; 100 6&quot;</td>
<td>500 3&quot; 100 5½&quot;</td>
<td></td>
</tr>
<tr>
<td>500 3½&quot; 500 5½&quot;</td>
<td>500 4&quot; 100 6&quot;</td>
<td></td>
</tr>
<tr>
<td>500 4&quot; 500 6&quot;</td>
<td>250 4½&quot; 100 7&quot;</td>
<td></td>
</tr>
</tbody>
</table>

Choose One Color or Random Assortment: Red, Blue, Green, Yellow, White, Orange, Black.

*#26 Prices on Request

**30 Kynar stripped 1" on each end. Lengths are overall. Colors: Red, Blue, Green, Yellow, Black, Orange, White. Wire packaged in plastic bags. Add 25¢/length for tubes.

#### HOBBY WIRE WRAP TOOL

- **BW 630 (Back Force)** $34.95
- **BT 2628 #26 Bit** 7.95
- **BT 2628 #26 Bit & Sleeve** 11.00

#### INDUSTRIAL WIRE WRAP TOOL

- **BW 928** $49.95
- **BW 928BF (Back Force)** 52.95
- **BW 928 #26 Bit & Sleeve** 4.95
- **BW 928 BF (Back Force)** 52.95
- **BW 928 #26 Bit & Sleeve** 4.95
- **Batteries & Charger** 11.00

#### ELECTRICAL INDUSTRIAL WIRE WRAP TOOL

- **EW 7D** $85.00
- **EW 7D BF (Back Force)** 92.90
- **EW 30 Bit & Sleeve** 29.50
- **EW 26 Bit & Sleeve** 29.50
- **Batteries & Charger** 11.00

#### INTERCONNECT CABLES

Ribbon cable connectors for connecting boards to front panels, or board to board.

- **Single Ended**
- **Double Ended**

#### OK PRODUCTS

- **WD 30** 50 ft. Wire Dispenser Red, White, Blue, or Yellow $3.75
- **WD-30-Tri** Tri Color Dispenser $5.50
- **R-30-TRI Refill for TRI Color** 3.75
- **INS 1416** 14 & 16 pin insertion Tool 3.25
- **MOS 40** 40 pin insertion Tool 7.50
- **EX-1** IC Extractor Tool 1.49
- **H-PCB-1 Hobby PC Board** 4.99
- **WSU 30 Hand Wrap/Unwrap/Strip Tool** 6.25
- **WSU 30M Same as WSU30 with Modified Wrap** 7.50

#### WIRE WRAP SOCKETS

- **8 pin** .40 .36 .34 .31 .27
- **14 pin** — .39 .37 .34 .32
- **16 pin** .35 .33 .31 .29
- **18 pin** .30 .28 .26 .24
- **20 pin** .25 .23 .21 .19
- **22 pin** .20 .18 .16 .14
- **24 pin** .15 .13 .11 .09
- **25 pin strip** .10 .08 .06 .04
- **40 pin** .05 .03 .01 .00

- **Gold 3-Level Closed Entry Design**

#### PAGE DIGITAL ELECTRONICS

135 E. Chestnut St. #5 Monrovia, CA 91016 (213) 357-5505

- **Solderless Breadboards**
  - **SK 10** $16.50
  - **2.2" x 6.5"**

- **LOGIC PROBE**
  - **PRB-1** $34.95
  - **Compatible with all Logic Families**
  - **10 Nsec pulse response**

- **Circle 298 on inquiry card.**

---

**BYTE July 1979 271**
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.

Inquiry No.  Page No.
1. A/B Computers 250
2. Advanced Access Group 154
3. Advanced Computer Products 237
4. Akaihin Automation 123
5. Altos Computer 23
6. American Square Computers 254
7. Anderson Jacobson 107
8. AP Products 59
9. ASAP Computer Products Inc 265
10. *Atlantic Analysis Corp 183
11. ATV Research 264
12. Automated Computer Software Service 85
13. Avery Label 254
14. Avionic Enterprises Inc 254
15. Basic Time 165
16. Beckman Enterprises 263
17. Biotech Electronics 189
18. *Bits Inc 137, 161, 236
19. Body Peripherals 95
20. Buss/Charles Forno 195
21. *BYTE Back Issues 234
22. *BYTE Books 37, 101-104, 192
23. *BYTE Industries 125
24. *BYTE WATS Line 258
25. California Computer Systems 18, 19
26. California Digital 241
27. CAP Electronics 232
28. Central Data 183
29. Chrysler Industries 225
30. *COMP/RINT! Computer Printers Int'l Inc 39
31. Computer (Div Intelligent Sys) 11
32. Computer (Div of Xerox) 143
33. Computer Components of South Bay 220
34. *The Computer Cookbook 41
35. *Computer Factory NY 245
36. *Computer Headware 207
37. *Computer Lab NJ 211
38. Computrend 8, 9, 160
39. Computrend 8, 9, 160
40. Computer MultiMedia 12
41. *Computer Packages Unlimited 254
42. *Computer Packages Unlimited 254
43. C G Engineering 258
44. Computhek 25
45. Computhek 25
46. Computhek 25
47. CT Micro Computer 94, 170, 212, 232
48. Cover Craft 190
49. Covox 213
50. Creative Computer Applications 215
51. Creative Software 128
52. Cromamek 1, 2
53. Cybernetics Micro Systems 216
54. Cybernetics Inc 183
55. Data Decisions 185
56. Data Discount Center 207
57. Datamation 97
58. Data/Print Publishing 231
59. DataSearch 216
60. Data/Speed 159
61. Data/Trans 267
62. Datec 188
63. Diablo (Div of Xerox) 145
64. Digital Pathways 143
65. DRC (CA) 177
66. DRC (TX) 255
67. Digital Data Systems Inc 258
68. Digital Service & Design 248
69. Digital Timing Devices 223
70. EASY SOFTWARE 215
71. 80-US Journal 177
72. Electrolinea 247
73. Electronic Control Technology 183
74. Electronic Systems 256, 257
75. Electronics Warehouse 243
76. EM/M/emi 207
77. Essex Publishing 218
78. Exidy Inc 32, 33
79. FMC Corp 185
80. Four Seasons Co 258
81. Allen Gailer 258
82. H Geller Computer Systems 177
83. Gemini 204
84. Ginric 243
85. Godbout Electronics 115
86. Graham Donian Enterprises Co
87. Grid 47
88. H & E Computerp 211
89. Hobby World 235
90. HUM Electronics 193
91. HUM Electronics 193
92. Industrial Micro Systems 127
93. International Peripheral Systems 65
94. INMAC 195
95. Integrals 221
96. Intercast Data Systems 29
97. Ithaca Audio 113
98. Ithaca Audio 209
99. Jade Co 266, 287
100. Jamesco 252, 283
101. King Flea 248
102. Med Hatter Software 151
103. *The Mail Mart 258
104. Marketline Systems 214
105. Matrox 185
106. Measurement Sys & Controls 131
107. Measurement Sys & Controls 254
108. Micro Ap 141
109. Micro Architect 222
110. Micro Datatax 174
111. Micro Diversions 91
112. Micro Focus LTD. 87
113. Micromol 203
114. Micro Mike's 191
115. Micro Pro International 89
116. Microsette 258
117. Micro Source 172
118. Microcomputer 201
119. The Micro Works 136
120. Mikos 264
121. Milton Bradley 51
122. Mini Computer Suppliers 175
123. Mission Control 131
124. Mission Control 133
125. Morrow/Thinker Toys 15
126. Morrow/Thinker Toys 31
127. Mountain Hardware 117, 156
128. National Small Computer Show 73
129. NECO 62
130. NECO 63
131. Netronics 155
132. Network Computer Systems 248
133. Newman Computer Exchange 269
134. NHX Olympic 232
135. North Star Computer 21
136. Ohio Scientific Instrument Co
137. OK Machine & Tool 139
138. OnComputing 81
139. Osborne & Associates 153
140. Owens Associates Inc 221
141. P & H Software 96
142. *P S Software House 248
143. Pacific Office Systems 268
144. Page Digital 271
145. PAIA Electronics 175
146. Per Com Data 78
147. Per Com Data 129
148. Per Com Data 259
149. Personal Computer Systems 118, 119
150. Pickles & Trout 234
151. Priority I 260, 261, 262
152. Pro Comp 260
153. Pro Comp 260
154. Pro Comp 260
155. Pro Comp 260
156. Quest Electronics 251
157. Quest Electronics 251
158. Quest Electronics 251
159. RACET Computers 234
160. Radio Shack Authorized Sales Center 254
161. RCA 75
162. Realty Software 258
163. The Recreational Programmer 222
164. RNB 93, 205
165. Rochester Data 187
166. Rothenberg Data Systems 200
167. S-100 Inc 221
168. S C Digital 211
169. St Jude 51
170. Michael Shayer Software 167
171. Shugart Associates 6, 7
172. Shugart Associates 6, 7
173. Smoke Signal Broadcasting 88
174. *Software Development & Training 178
175. *SSM 30
176. Southwes Technical Products Corp CII
177. *Stirling Beker 202
178. SubLOGIC 218
179. Sunny International 268
180. Sybex Inc 157
181. Synchro Sound 83
182. Tarbell Electronics 67
183. Technical Systems Consultanta 111
184. Technology Systems South 148
185. 3M Company 27
186. Robot Time Graphics 55
187. Tera Systems Inc 254
188. *Trade of Industrial Products Inc ITIP 5
189. TransData 213
190. Trparsen 233
191. TRS-80 Software Exchange 147
192. Verbatim 21
193. Video World 221
194. *Vista Computer 213
195. *Vista Computer 213
196. VR Data 263
197. Voltage/Time Graphics 26
198. WAU Electronics 126
199. Wangcom 264
200. Wainwright Corp 248
201. *X-Comp 254
202. X-Comp 254
203. *Zig Systems 248

*Correspond directly with company.

BYTE's Ongoing Monitor Box

April BOMB Reaches for the Stars

Results of the April BOMB put simulation at the top of the charts.


272 July 1979 © BYTE Publications Inc
Graham-Dorian’s Integrated, On-Line Programs Are Fast, Efficient, And Easy To Use.

Rely on Graham-Dorian, a full-line computer software manufacturer, for sophisticated programs — the most detailed on the market today. They’re ready to go to work immediately or to be tailored for even more specific needs.

On-line capabilities enable you to make a single entry and update all affected files. An inquiry into a file at any time provides up-to-date information — no batching or sorting of input data.

The programs are easy to use. Messages on the video display guide you each step of the way. Programs make use of indexed sequential and chained files for fast and convenient retrieval of data with efficient use of disk space.

Order in standard eight-inch disk either double or single density, or various mini-floppy formats. Each program contains a free CBASIC-2, plus user's manual and hard copy source listing.

- **Accounts Receivable** — Records invoices, prepares statements and trial balance reports, etc. Automatically reports aging of accounts in periods of 30, 60, and 90 days, with each item listed separately.

- **Accounts Payable** — Vendor lookup and change, entering vendor invoices, writing checks (many options), cash flow analysis, accounts payable check register, and vendor list. Ideal for analyzing expenditures by vendor and by due date.

- **General Ledger** — Includes lookup and change, making journal entries, trial balance, transaction register, chart of accounts, financial statements, and monthly closing.

- **Job Costing** — Provides work order lookup, enters labor transactions, material set-up, progress report of hours, labor distribution report, weekly labor reset, actual versus estimated cost per job.

- **Inventory** — Can be connected with cash register for point of sale inventory control. Number of on-line items limited only by disk space available.

- **Cash Register** — Creates daily sales reports containing information on gift certificates, paidouts, overrings, refunds, and how much in each category a salesperson sold.

- **Payroll** — Handles 100% of all necessary payroll functions including state income tax tables for your state. Ideally suited for both large and small companies.

- **Apartment** — Said one user, “Obviously, this was developed by apartment owners.” The package fills virtually all the needs of apartment owners and managers. Ideal for projects with 75 units or more.

- **CBASIC-2** — The most comprehensive and powerful commercially oriented BASIC available today. Enhancements over CBASIC-1: integer variables, multiple line functions, CHAINing with COMMON variables, additional predefined functions, etc. Comes free with any package, or purchase separately.


See your GDESS dealer or send for information packet and sample runs.
Ohio Scientific has taken its standard Challenger III computer and married it to the new Shugart 29 Megabyte Winchester Drive. The result is the C3-C. This new microcomputer now fills the vacuum that existed for computer users who need more mass storage capability than floppies can offer—yet until now, could not justify the additional cost of a larger capacity hard disk computer such as our C3-B 74 Megabyte disk system.

Winchester Technology

Winchester hard disk drives offer small business and professional computer users the logical solution to mass storage problems that are beyond the capability of floppy disks. In addition, Winchester disks feature a track seek-time that is much better than floppies and because they spin at eight times the rate of floppies, Winchesters have a shorter latency. Both of these points reflect one remarkable speed advantage Winchester disks have over floppies.

Coupled to the Challenger III Computer

Ohio Scientific's award winning Challenger III computer is a classic. It is the only computer series that utilizes the three most popular microprocessors—6502A, 68000 and Z-80. This tremendous processor versatility enables one to utilize seemingly endless selection of quality programs available from Ohio Scientific's software library as well as from many independent suppliers.

And Advanced Software

For instance, there are single user, multi-user and network operating systems. A complete turnkey small business package, OS-AMCAP provides accounts receivable, accounts payable, disbursements, cash receipts, general ledger, etc. OS-CP/M offers a complete FORTRAN and COBOL package. And there is WP-2, a complete word processing system. For information management, OS-DMS, features an advanced file handling system and program library that simplifies information storage and recall and routinely performs tasks which usually require special programming on other systems.

Ohio Scientific does it again

Yields the Microcomputer of the Future

With an eye toward the future, the C3-C, like all other Challenger III's was designed with provisions for future generation 16 bit microprocessors via plug-in options. There are ten open slots for lots of I/O and multi-user operation. Truly, the Ohio Scientific C3-C is a computer with a future.

The new C3-C computer with 29 Megabyte Winchester Hard Disk.