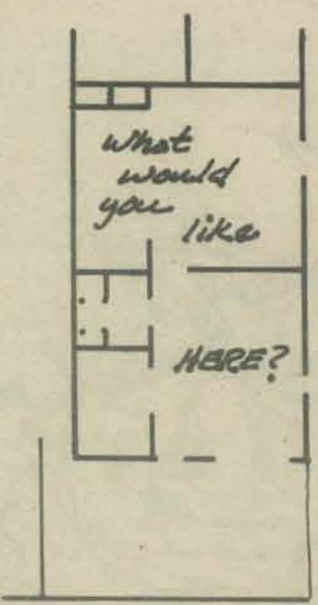
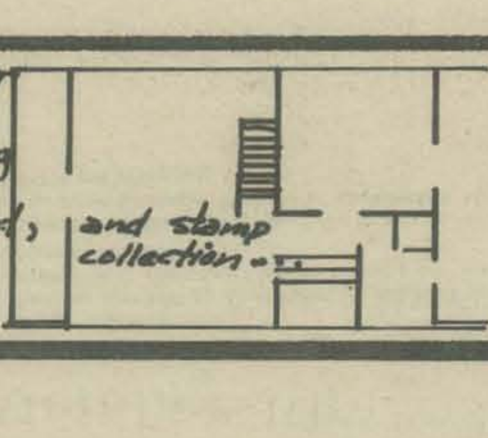
PEOPLE'S COMPUTER COMPANY

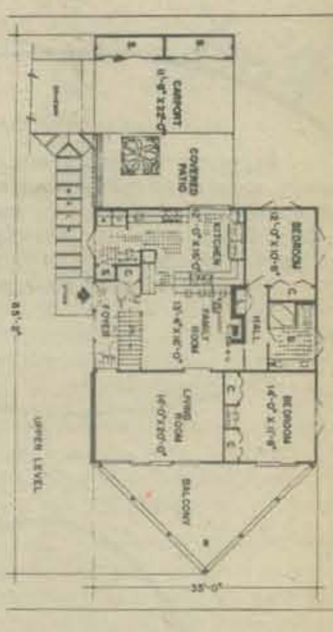
VOLUME 3 MARCH '75 NUMBER 4



what would you like HERE?

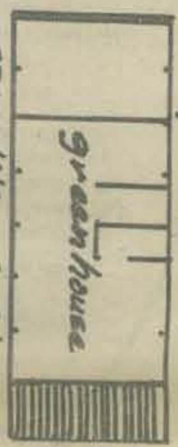


C.G. DEGROOT ... storing and retrieving data on a tape, record, and stamp collection.

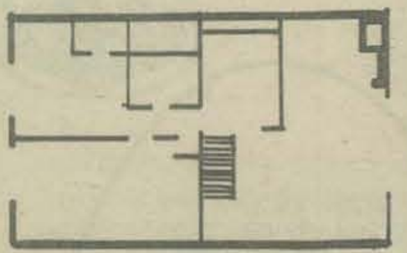


There is only one more issue this year so...

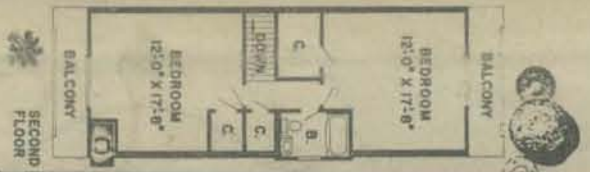
RENEW NOW

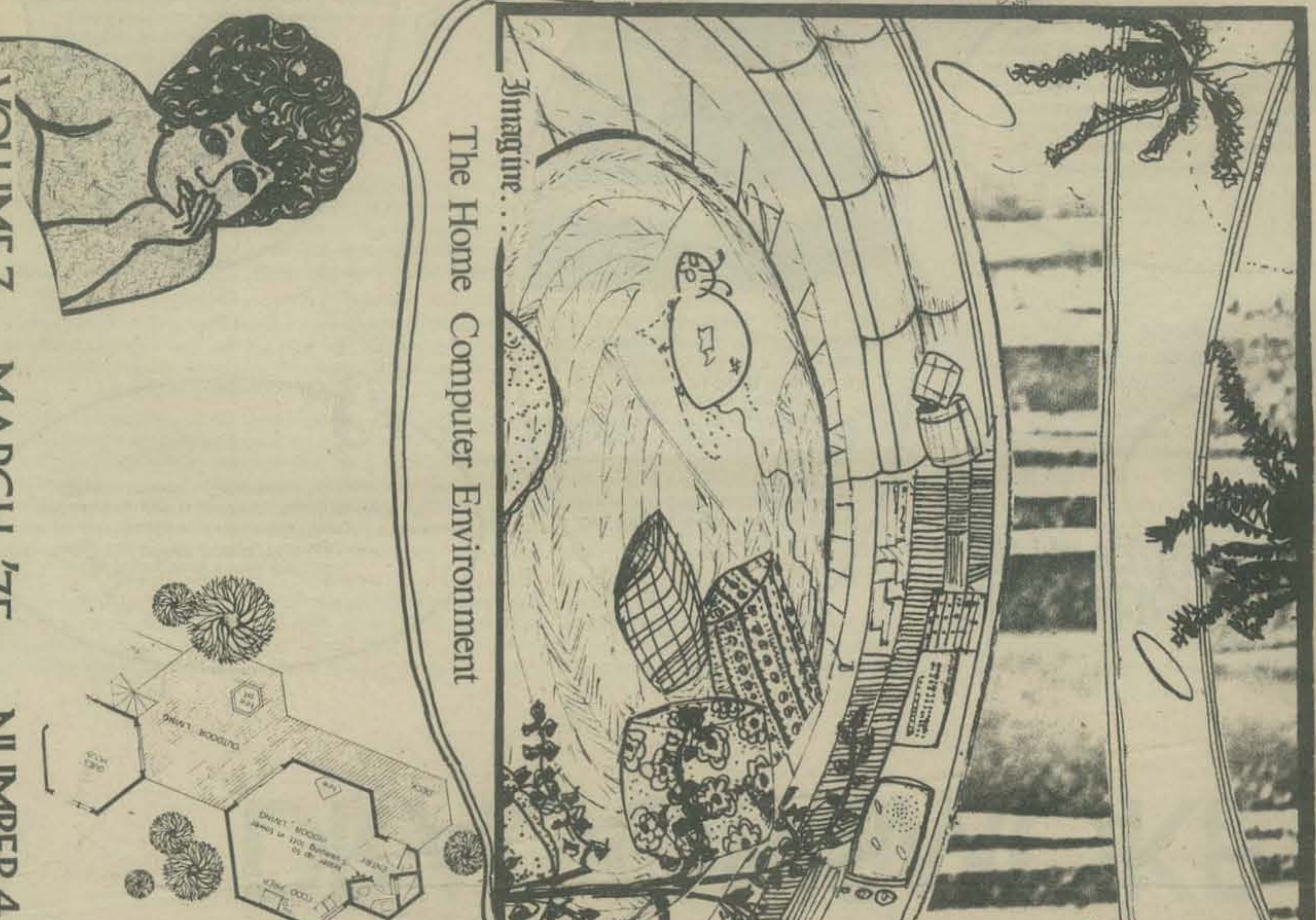


D. BERGGREN ... control an elaborate existing burglar alarm system; temperature and humidity control in his greenhouse lawn sprinkler system, and for bookkeeping...



F. STAUBER ... speed, drift, and heading calculations for airplanes and to receive and send morse code ...

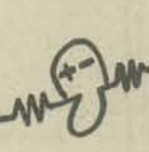




Imagine... The Home Computer Environment

DRAGONSMOKE

The future came a little early this year. The HOME COMPUTER exploded all over the place. Everything I thought I knew about computers is *obsolete.* Watch the PCC Dragon struggle to keep up!



DRAGONFLIGHT

Lessa wondered how she knew the golden dragon's name, and Ramoth replied: Why shouldn't she know her own name since it was hers and no one else's? And then Lessa was lost in the wonder of those magnificiently expressive eyes.

Oblivious to the descending bronze dragons, oblivious to the presence of their riders, Lessa stood caressing the head of the most wonderful creature of all Pern, fully prescient of troubles and glories, but most immediately aware that Lessa of Pern was Weyrwoman to Ramoth the Golden for now and forever.

Sigh . . . now that's the way to treat a dragon!

DRAGONFLIGHT
by Anne McCaffrey
Ballantine Books, Inc.
201 E. 50th St.
New York, NY 10022

The ALTAIR 8800

Our friendly unimpeachable source tells us that 2600 ALTAIR 8800 home/school minicomputers will be delivered during the first 3 months of 1975. And . . . he continues . . . MITS will soon be delivering ALTAIRs at the rate of 1500 per month. Amazing!

Our very own ALTAIR is up and running — tell you more about it in the next issue. In the meantime, *if you have acquired an ALTAIR, please write!*

MITS INC.
P.O. BOX 8636
ALBUQUERQUE, NEW MEXICO 87108

MITS has announced an ALTAIR USER'S CLUB. They say —

Purposes:

1. The development of a comprehensive library of software for the *Altair Computer.* This library will, of course, be made available to all members of the User's Club.
2. Provide *Altair* Users with efficient, *priority* service. This service will range from ordering *Altair* options to free technical consultation.
3. Provide Users with useful, up-to-date information about the *Altair, Altair* software, and computers in general through a Club newsletter.

Membership:

Everyone whorbuys an *Altair Computer* will get a free, one year Membership to the User's Club (including a one year subscription to the Club newsletter).

Associate Memberships are available to anyone who does not own an *Altair Computer.* The cost of an Associate Membership is $30.00 per year.

Simulation/ Gaming/News

S/G/N zooms in, zooms out on the world of simulations, games and simulation games. Each issue has news, philosophy and a complete game to play.

Simulation/Gaming/News is issued six times a year (early in January, March, May, July, September and November). Subscriptions are $6 a year, and checks (not purchase orders) must accompany subscription requests. Zip codes also should be specified. Advertising rates and specifications are available upon request. All communications should be addressed to S/G/N, Box 3039, University Station, Moscow, Idaho 83843.

CALCULATORS
IN THE
CLASSROOM

by Marvin Grosswith in
Datamation (March '75)
Technical Publishing Co.
1301 South Grove Ave.
Barrington, IL 60010

OK, OK — we'll start do doing calculator stuff next issue.

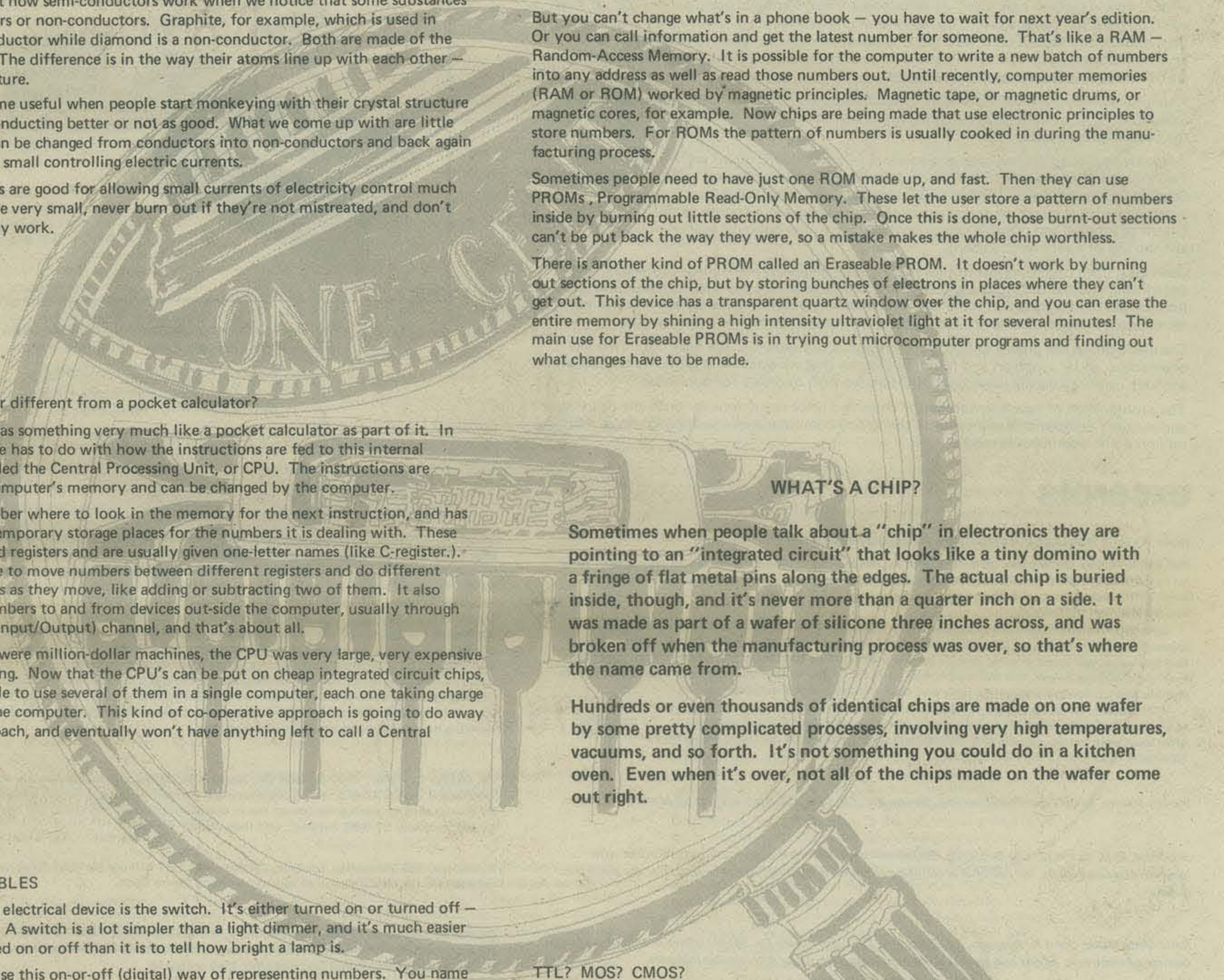
Dear PCC:

I am offering computer peripheral equipment to hobbyists interested in developing a sophisticated system at unbelievable savings. I am not a commercial dealer: I am a hobbyist who was able to purchase this equipment at an attractive price. Contact D. Miller, 1191 Risa Place, Santa Ana, Ca. 92705.

Write for a list of stuff and prices

Chip Talk

by Lee Felsenstein



SEMICONDUCTOR

When electricity will pass through a substance without creating much heat, the substance is called a conductor of electricity. If electricity won't pass through the substance at all, it's a non-conductor. Halfway between are the semi-conductor elements, usually metals with strange names.

We can get a clue about how semi-conductors work when we notice that some substances can be either conductors or non-conductors. Graphite, for example, which is used in pencil leads, is the conductor while diamond is a non-conductor. Both are made of the same thing – carbon! The difference is in the way their atoms line up with each other – called the crystal structure.

Semi-conductors become useful when people start monkeying with their crystal structure to get the substance conducting better or not as good. What we come up with are little dabs of metal which can be changed from conductors into non-conductors and back again under commands from small controlling electric currents.

Semi-conductor devices are good for allowing small currents of electricity control much larger currents. They're very small, never burn out if they're not mistreated, and don't make much heat as they work.

CPU

What makes a computer different from a pocket calculator?

Well, every computer has something very much like a pocket calculator as part of it. In fact, the read difference has to do with how the instructions are fed to this internal calculator, which is called the Central Processing Unit, or CPU. The instructions are stored in part of the computer's memory and can be changed by the computer.

The CPU has to remember where to look in the memory for the next instruction, and has to be able to provide temporary storage places for the numbers it is dealing with. These storage places are called registers and are usually given one-letter names (like C-register.). The CPU has to be able to move numbers between different registers and do different things to these numbers as they move, like adding or subtracting two of them. It also has ways of getting numbers to and from devices out-side the computer, usually through what's called the I/O (Input/Output) channel, and that's about all.

Back when computers were million-dollar machines, the CPU was very large, very expensive and had to do everything. Now that the CPU's can be put on cheap integrated circuit chips, it's becoming reasonable to use several of them in a single computer, each one taking charge of a different part of the computer. This kind of co-operative approach is going to do away with the big-boss approach, and eventually won't have anything left to call a Central Processing Unit.

BITS, BYTES and NIBBLES

Just about the simplest electrical device is the switch. It's either turned on or turned off – there is no in between. A switch is a lot simpler than a light dimmer, and it's much easier to tell if a lamp is turned on or off than it is to tell how bright a lamp is.

Almost all computers use this on-or-off (digital) way of representing numbers. You name these choices A and B, Herman and Joan, or give them numbers like 1 and 0 (that's what mathematicians like best). Then think up a code which can make larger numbers out of these little choices which are called bits. (You have to admit that a number which can only choose between two things is pretty small, and deserves to be called a bit.)

There are a lot of possible codes which can use bits to make up larger numbers, and computers use several different codes (BCD, Excess-3, Gray Code, and others, each with its own reason for being used). But the code which uses the smallest number of bits to make a larger number is called Binary, and it is used in almost all computers.

Letters and symbols like those on a typewriter can be expressed in binary code, too. You just give each letter a number and use binary code to translate that number into a bunch of bits. When you do this, you have to keep the bunches of bits which represent each letter separate from each other, so you know where the next letter begins.

A bunch of bits which has to be kept separate from the next bunch is called a Byte. Usually it is 8 bits long. The name is pronounced "Bite" and in one computer where numbers were handled in chunks of four bits the designers called those chunks "nibbles."

ROM? RAM? PROM?

These are kinds of memories. (That's what the M stands for – memory). Every computer has a way of remembering numbers and instructions. If the list of numbers or instructions will never change, it can be stored in a ROM, or Read-Only Memory. A ROM is like a phone book – the computer looks for a certain name and gets a number back. The name is called the address, and like everything else in a computer, it's make up of numbers.

But you can't change what's in a phone book – you have to wait for next year's edition. Or you can call information and get the latest number for someone. That's like a RAM – Random-Access Memory. It is possible for the computer to write a new batch of numbers into any address as well as read those numbers out. Until recently, computer memories (RAM or ROM) worked by magnetic principles. Magnetic tape, or magnetic drums, or magnetic cores, for example. Now chips are being made that use electronic principles to store numbers. For ROMs the pattern of numbers is usually cooked in during the manufacturing process.

Sometimes people need to have just one ROM made up, and fast. Then they can use PROMs , Programmable Read-Only Memory. These let the user store a pattern of numbers inside by burning out little sections of the chip. Once this is done, those burnt-out sections can't be put back the way they were, so a mistake makes the whole chip worthless.

There is another kind of PROM called an Eraseable PROM. It doesn't work by burning out sections of the chip, but by storing bunches of electrons in places where they can't get out. This device has a transparent quartz window over the chip, and you can erase the entire memory by shining a high intensity ultraviolet light at it for several minutes! The main use for Eraseable PROMs is in trying out microcomputer programs and finding out what changes have to be made.

WHAT'S A CHIP?

Sometimes when people talk about a "chip" in electronics they are pointing to an "integrated circuit" that looks like a tiny domino with a fringe of flat metal pins along the edges. The actual chip is buried inside, though, and it's never more than a quarter inch on a side. It was made as part of a wafer of silicone three inches across, and was broken off when the manufacturing process was over, so that's where the name came from.

Hundreds or even thousands of identical chips are made on one wafer by some pretty complicated processes, involving very high temperatures, vacuums, and so forth. It's not something you could do in a kitchen oven. Even when it's over, not all of the chips made on the wafer come out right.

TTL? MOS? CMOS?

Names of integrated circuit types have gotten so long that people who use them all the time use abbreviations. MOS, for instance, stands for Metal Oxide Semiconductor. What it means to people who use it is that it's a process which allows a lot more stuff to be cooked onto a chip than any other process (yet). So MOS is good for making LSI (Large Scale Integration) chips. LSI packs a whole lot of stuff on a single chip. You never get anything for free, though, so MOS chips are slower operating than the other kind.

That other kind is called bipolar logic, and the main kind that's used today is TTL, or transistor-transistor logic. It's not good for LSI because it makes too much heat and takes up too much space. It is used mainly for MSI (Medium-Scale Integration) and SSI (guess what the S stands for) devices.

CMOS will someday be as cheap, although not as fast as TTL. CMOS stands for Complementary MOS. It's a way of making MOS chips that use almost no power, and are faster than MOS. CMOS is not very good for LSI, but great for uses like electronic watches where power has to come from a battery.

A word of caution – MOS chips are incredibly sensitive electrically. The amount of static electricity you can generate from just wearing nylon clothing is enough to destroy a chip internally if you touch it. This is especially true of microcomputer chips and their memory chips. So don't be too quick to handle a MOS chip if you aren't sure about the safety precautions!

micro-8 newsletter

The Micro-8 User Group

by *Harold L. Singer*
Cabrillo Computer Center
Cabrillo High School
Lompoc, Ca. 93436
805-733-3501

Radio-Electronics magazine has published two articles that have launched a new electronic components market, the hobby computer. This started with RE's Sept. 1973, "TV Typewriter" article by Don Lancaster. It described a Video Terminal using a home TV set that could be built for under $150. The response was overwhelming! One month after publication, most of the special parts were back ordered for months. Many thousands of these have been built by hobbyists and there is such continued interest that several companies are offering complete component parts kits.

A natural followup article, "The Mark-8 Minicomputer", by John Titus appeared in RE in July 1974. This was a slightly modified implementation of the INTEL MCS-8 system using the 8008 microprocessor.

Response to this article has been phenomenal. Over 2000 PC board sets have been sold. Two months after the article appeared a letter to the editor appeared in RE offering the Cabrillo Computer Center staff as a clearing house for a MARK-8 User Group. In the first two months we acquired an active group of 220 members in response to this single letter. Extent of interest and possible component sales from the hobbyist market appears to be just the top of a very large iceberg.

Users vary from beginning electronics hobbyists to engineers and college professors, and a surprising number of Doctors and Dentists. Application for the 8008 systems vary widely from simple hobby projects, home environmental and security systems, educational and recreational systems, to elaborate systems to be used in the builder's occupation.

The user's group has grown to 500 members and is still growing. We have published five newsletters so far. Interest and response dictates that we establish a formal organization and very carefully decide what kind of future we wish to create for ourselves.

The proliferation of new microprocessor chips and price reductions for both the processors and memory components will result in the hobby computer movement producing a multi-million dollar components market.

The MARK-8 is a complete minicomputer based on the INTEL 8008 microprocessor. For info, write:

Radio Electronics - Micro-Computer
P. O. Box 1307 - Radio City Station
New York, New York 10019

and join the MICRO-8 User's Group!

excerpts

HOW TO GET RELIABLE HARD COPY PRINTOUT FOR $125

Bob Cook, Wilcox Enterprises, 25 W 178 39th St., Naperville, Il, 60530, provided us with what I think is the most exciting information to hit the hobby computer group yet. Enclosed is a flyer offering Creed TTY's manufactured in England by a subsidary of IT&T in 1964 for a specialized order processing system used by Spiegal.

Bob has done a beautiful job of preparing his flyer and it will take many re-readings before you appreciate all of the features these offer. He also provides a MARK-8 interface kit with the unit so that the buyer will know exactly how to get the unit running, has PROM's available with a keyboard monitor described below, and will supply free parts for 1 year so that you can keep them running. He also has service documentation available and sells each unit with a detailed manual.

A reasonably quiet and modern, 10 CPS, hard copy device and keyboard with a paper tape punch as a bonus for $125. That has to be a dream come true for the MARK-8 participant. To make the offer seem even better, circuits and PROMS will soon be available for using the Creeds with the Mil Monitor-8 described in the next section.

Harvey Heintz, 9730 Townline Diversion, Surrey B.C. CAnada V3S 4N7 will use his Mark-8 for games.

Jim Mehl, P. O. Box 632, Los Gatos, Ca. 95030 will use his Mark-8 as an intelligent terminal with local editing capabilities, for personal accounting and record keeping, and as a controller for electronic games.

Solid State Music, John R. Burgoon, 1222 Carolyn Dr., Santa Clara Ca. 95050 is into design of computer-control of electronic organs and wants to communicate with other with similar interests.

Carl K. Zettner (W5HFG), 108 Moss Drive, San Antonio TX 78213 points out that the Datapoint 2200 has the same instruction set with only four exceptions so Datapoint software may be useable with minor modifications.

Ron Angstadt, Rd. 3 Box 281, Kutztown Pa. 19530 plans to use his Mark-8 for home security, TV games and inventory on food supplies.

William R. Mains, 139—17th St., Pas Robles, Ca. 94401 is waiting till the right time to build up an educational system for his high school using either an 8008 or 8080 microporcessor.

Mark Spohr, 527 Lafayette Ave., Cincinnati OH is a senior Med student at Univ. of Cincinnati and is interested in using the Mark-8 for diagnosis, treatment, drug dosage interactions, and health systems planning. (Several doctors have gotten together thru the newsletter.)

D. Bergren, 9207 SW 43rd Ave., Portland OR 97219 will use the Mark-8 to control an elaborate existing burglar alarm system, for temperature and humidity control in his greenhouse, for a lawn sprinkler system, and for bookkeeping.

Bro. R. W. Harris, Monsignor Farrell HS, 2900 Amboy Rd., Staten Island NY 10306 plans to use the Mark-8 as a teaching tool for assembly language.

Joseph A. Cimmino, 19304 Richwood Court, Brookeville, Md. 20729 is promoting a Washington DC User's Group and intends to set up an 8008 and 8080 software distribution house. The Group's resources include Mark-8's, Intellec 8's, MIL MOD-8's, big computers, etc., and they are contemplating big things for the future. They intend to have one each of the semistandard devices the User Group establishes so they can do software modification from one format to another. They have purchased the INTEL User Library. They will soon set up a telephone communications network using Mike Hayes' (MNH Electronics) modems and Joe is even suggesting use of the ham satellite network for national distribution of software. Maybe we can get them to write up what they have going for the next newsletter. Joe is doing the group a service by putting the roster into machine readable format. Contact him if you want to buy a complete set of stick-on mailing lables. He will soon mail out an elaborate survey form asking participants dozens of questions regarding what they are doing and waht they want to do in the future. He will compile this info and make it available to all of us so please take filling it out seriously. He is also working on a super 8008 floating point package.

Dr. George L. Haller, 1500 Galleon Dr., Naples Fla. 33940, has a Scelbi w/2K of memory, Tape interface & model 32 RO. He's programming it for golf handicaps (which consists of accepting 20 scores; sorting; adding the lowest ten in double precision; subtracting the course rating; dividing by ten; multiply by 0.85 (whew!), and then output the result.

F. Stauber, 183 Woodville, Toronto, Canada M4J 2R4 will use his Mark-8 for speed, drift, and heading calculations for airplanes and to receive and send Morse Code.

Charles F. Douds, 381 Poplat St., Winnetka, Il 60093 is deeply involved in electronics in model railroading. He is connected with the Garfiled-Clarendon Model Railroad Club and anyone with similar interests should contact Charles and find out what they are up to.

John C. Neves, 930 Palo Alto,Ave., Palo Alto, Ca. 94301 is interested in Morse Code translation and wants to try digital filtering algorithms.

C. G. Degroot, 155 W. 81 St., No. 4J, NY, NY 10024 is interested in storing and retrieving data on a tape, record, and stamp collection and wants a good cassette circuit.

MARK-8 MODS, A $80 VIDEO TERMINAL, & ANOTHER CASSETTE UNIT THAT WORKS!

The Digital Group, P. O. Box 6528, Denver, Co. 80206 has an exciting package of information that will soon be available. The best way to describe it is to extract paragraphs from letters they have sent:

"A guy named Dr. Robert Suding called us about showing his complete Mark-8 to those in the Denver area and about 15 showed up. Dr. Suding has a doctorate in systems analysis, works for IBM and his latest development project is in graphics. He is also one of the nation's leading developers of slow-scan TV. He has made extensive MODS to his Mark-8 including pluggable boards, octal readout, large power supply, and keyboard data entry. He spent a ½ day designing an FSK cassette interface after rejecting other circuits due to drift. Result – not one bit error in over 2 months using a K-Mart $30 el-cheapo. His crowning achievement is a TV character generator that uses 1101 RAM (almost instantaneous update) and displays upper and lower case, numbers, symbols and the greek alphabet and that can be built for under $80."

WHAT'S HAPPENING

Word has it that MITS is about to lay an ALTAIR 8800 on People's Computer Company. Which has got to prove that you are doing something right. Now if I can only convince you that you ought to use all that talent youhave at PCC to write a good cross-assembler in BASIC, all the grateful game playing multitude will be able to include home style computer building in their futures. (Come on you guys, he needs convincing. Write to him!) Meanwhile I won't tell them that you have plans to provide a BASIC language processor for the 8800 system. There is plenty of talent at PCC to do the necessary.

Incidentally, I'm glad to hear that you intend to give Hal Singer a hand with the MARK-8 user group. He has done a fine job, and a whole lot of us owe him a debt of gratitude.

The other day Fred Moore dropped by. Fred is one of the people who tried to get an information network up and running. His idea was to get everyone to register under their particular interest area. Fred then planned to distribute the roster to everyone on the roster, and therefore get people the means to get in touch with other people. To make a long story very short it was too difficult to do with the means Fred had at hand. Now Fred has launched a local Homebrew Computer Club.

I've been following the articles on the CRT display that "The Computer Hobbyist" is publishing. My opinion is that this is the best CRT display system that I've seen. It is reminiscent of the method used in the PDP-12/LINC-8 CRT displays. It is different than the TVT concept in that the beam does not scan, but it is positioned by X and Y commands. In short it handles more like a plotter system. That is you can beam on (intensify point)(pen down) or beam off (lower intensity)(pen up). This yields a lot more flexibility to the character of the displays that are possible. Figures, lines, curves, and kaliedoscope effects are relatively easy to do. The method has been around long enough to be good, practical, and versatile. They are describing the hardware and the software and are publishing routines that draw the chess board and chess pieces. If this turns you on, their address is:

The Computer Hobbyist
Box 295
Cary, North Carolina 27511

Six inflated American dollars will get you a years worth of very good stuff. They are covering the area of logic symbols, surplus markets, and the letters from some of their subscribers are describing some of the equipment that has traditionally been available, like TTY of all kinds sorts, styles and descriptions. They also have done an article on interfacing the TTY Model 15 to a home computer system. Since that was the machine that I used in my system (version one) I was interested in that very much. I still feel that for $50 – $75 a Model 15 can't be beaten. I think that it is a whole lot less work to rewire this machine than to wire a TVT thingee from scratch. If you ever do undertake this project, you will learn all there is to know about TTY signals and it does make stepping up to an ASCII device a great deal easier to understand after you have made a Model 15 work. You can, incidentally still use ASCII for the other devices that you have hung on your system and go through a short piece of translation code just before you need to print anything. If you are being held up on account of you don't have a print type of device, you might ask the North Carolina boys for the back issues and take a look. There is information as to where to go and what to buy. There are outfits around Chicago (if my memory serves me) that buy and sell this type of gear. They cater to the radio TTY boys. There is a variety of Baudot (five level code) machines that are 110 baud machines.

The talks that I've had with the men who count in this hobby all agree that the home computer will come into it's own when the machinery and the software can handle a common garden variety color TV set. Lee Felsenstein of:

LGC Engineering
1807 Delaware Street
Berkeley, Ca. 94703

with the others (apparently) are working on a somewhat elaborate but effective distributed processor. They will hate me for describing their effort that way, and I personally would prefer to have a system like the one they are describing. It will however, be more expensive to get up than a simple conventional computer system. But when it is together it will be a pip! I would suggest that you get their system description before you get very far down line, and see if you can use their ideas. They also have some hardware for sale at good prices. But the main thing that LGC Engineering is doing is that they are publishing a series of papers on what they call The Tom Swift Terminal. They have done much thought and work on the problem, and my guess is that they will be among the first to solve the many problems involved in handling a TV type display, while you massage and make ready the data you are going to display. Their system will operate more like the more sophisticated and expensive display terminals in commercial use, but they will be built up of inexpensive components and based on 8008 microprocessors, or processors of that type. At any rate, they are on to the problems like how often to scan and how you can supply the TV with info that is being updated by other devices that are manipulating the data. The essential difference as I understand their design, is to have a processor in each of the devices that hang on the data lines. I know for a fact that this is the very scheme that is used at Ames Research Center of one of the pilot simulation systems that has an Evans and Sutherland line drawing system hung on it.

Somebody else who seems to be going gang busters is Carl Helmers, Jr., who is otherwise known as:

MP Publishing Co.
Box 378
Belmont, Ma. 02178

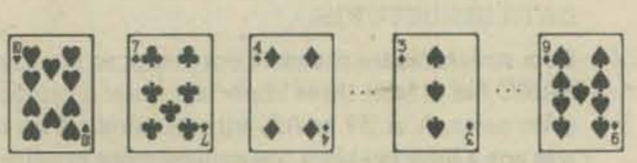
Is publishing a series of booklets covering the construction of what he calls "The Experimenter's Computer System" or ECS-8 for short. Send him $21 for 12 issues of this system. He is developing it in serial form in the booklet. It is getting better and better as he goes along. He includes some choice coding charts which give instruction times and other valuable information that you would compile for yourself if you had the time. He has designed an LED display and hung it on an output port so you can ship bits out there and take a look at them if you want to. He is into a really stable clock, and has published the best tutorial description of UARTS that I've ever seen in print. The software to run his stuff is also in the booklet and the feeling that you come away with after reading his stuff a little while is that with just a little more effort and a few additional check lists, he is putting out the paperwork of a heathkit. Other good stuff and ideas that he has come up with are that he has what he calls a kluge assembler system. It is a pre-printed form with the byte numbers and the space to fill in the instructions as you code them. It is precisely what I have been painstakingly doing for myself for several weeks now. He wants a nickle a sheet for them and instructions are written on the heel (so to speak.) He is also aware of the software that is/or will be necessary to get all the stuff playing and he is hard at work developing that too. It is a very creditable effort.

I notice that I have passed over Hal Singer of the MARK-8 User Group rather lightly, but that was unintentional. Hal has done a masterful job of editing the letters that he has been getting. The only criticism of that valiant effort to stay afloat of a sea of paperwork is that those $%*&$$ ditto sheets are impossible to read. But he is getting a lot of info on devices and techniques, people and places published.

One final note. Those of you who have already started your 8008 system need not panic or dispair becasue MITS is coming out with a better goodie. If you can finish your system as you originally set out to do, go right ahead. I personally think that the 8080 for all of it's obvious advantages ia a lot more computer than I could use this year at home. By the time I have gotten to the end of developing my 8008 system to the full, memory costs, easier interface techniques and devices will be available. And more important I will know better just what it is that I intend to do with the system. Understand that I'm not putting down the Altair 8800, far from it. But my experience is that it takes a long time to develop anything that is really worth while, and I for one am not going to junk what I've started until I have squeezed out all that I can from a really inexpensive system. Memory cost is still the big hang up and it will continue to be for some time yet. Besides, a clipped canary at a tom cat's convention has a better chance than I have of convincing my lady that I need a disk system for the living room.

If you have gotten anything valuable out of this little chat, write me, Gordon French, care of PCC. Yes, I can deal with critics too! Let me know what you'd like to see and if we can sandwich it in we'll do it.

Gordon French



AMATEUR COMPUTER USERS GROUP
HOMEBREW COMPUTER CLUB . . . you name it.

Are you building your own computer? Terminal? T V Typewriter? I/O device? or some other digital black-magic box?

Or are you buying time on a time-sharing service?

If so, you might like to come to a gathering of people with likeminded interests. Exchange information, swap ideas, talk shop, help work on a project, whatever . . .

This simple announcement brought 32 enthusiastic people together March 5th at Gordon's garage. We arrived from all over the Bay Area--Berkeley to Los Gatos. After a quick round of introductions, the questions, comments, reports, info on supply sources, etc., poured forth in a spontaneous spirit of sharing. Six in the group already had homebrew systems up and running. Some were designing theirs around the 8008 microprocessor chip; several had sent for the Altair 8800 kit. The group contained a good cross section of both hardware experts and software programmers.

We got into a short dispute over HEX or Octal until someone mentioned that if you are setting the switches by hand it doesn't make any difference. Talked about other standards: re-start locations? input ports? better operating code for the 8080? paper tape or cassettes or paper & pencil listings? Even ASCII should not be assumed the standard: many 5 channel Model 15 TTYs are about and in use by RTTY folks. Home computing is a hobby for the experimenter and explorer of what can be done cheaply. I doubt that standards will ever be completely agreed on because of the trade-offs in design and because what's available for one amateur may not be obtainable for another.

Talked about what we want to do as a club: quantity buying, cooperation on software, need to develop a cross assembler, share experience in hardware design, classes possibly, tips on what's currently available where, etc. Marty passed out M.I.'s Application Manual on the MF8008 and let it be known that he could get anything we want. Steve gave a report on his recent visit to MITS. About 1500 Altairs have been shipped out so far. MITS expects to send out 1100 more this month. No interfaces or peripherals are available until they catch up with the mainframe back orders. Bob passed out the latest PCC and showed the Altair 8800 which had arrived that week (the red LEDs blink and flash nicely). Ken unboxed and demonstrated the impressive Phi-Deck tape transport.

What will people do with a computer in their home? Well, we asked that question and the variety of responses show that the imagination of people has been underestimated. Uses ranged from the private secretary functions: text editing, mass storage, memory, etc., to control of house utilities: heating, alarms, sprinkler system, auto tune-up, cooking, etc., to GAMES: all kinds, TV graphics, x - y plotting, making music, small robots and turtles, and other educational uses, to small business applications and neighborhood memory networks. I expect home computers will be used in unconventional ways--most of which no one has thought of yet.

We decided to start a newsletter and meet again in two weeks. As the meeting broke up into private conversations, Marty held up an 8008 chip, asked who could use it, and gave it away!

Fred Moore
2100 Santa Cruz Ave.
Menlo Park, Ca. 94025

Announcement:

Texas Instruments Learning Center is presenting an early morning home television series, April 15 - 18, on "Introduction to Microprocessors." Check your local newspaper for time (6:00 AM) and channel.

BUILD YOUR OWN BASIC

by Dennis Allison & Others

A DO IT YOURSELF KIT FOR BASIC??

Yes, available from PCC with this newspaper and a lot of your time. This is the beginning of a series of articles in which we will work our way through the design and implementation of a reasonable BASIC system for your brand X computer. We'll be working on computers based on the INTEL 8008 and 8080 microprocessors. But it doesn't make much difference – if your machine is the ZORT 9901 or ACME X you can still build a BASIC for it. But remember, it's a hard job and will take lots of time particularly if you haven't done it before. A good BASIC system could easily take one man six months!

We'd like everyone interested to participate in the design. While we could do it all ourselves, (we have done it before) your ideas may be better than ours. Maybe we can save you, or you can save us, a lot of work or problems. Write us and we'll publish your letter and comments.

DIRECT MODE?

Some kind of "desk calculator" mode of operation would be nice. At least, we would like to be able to look at and set different variables in a program and restart execution at any given point. This feature makes it easier to find and gently terminate the existence of "bugs."

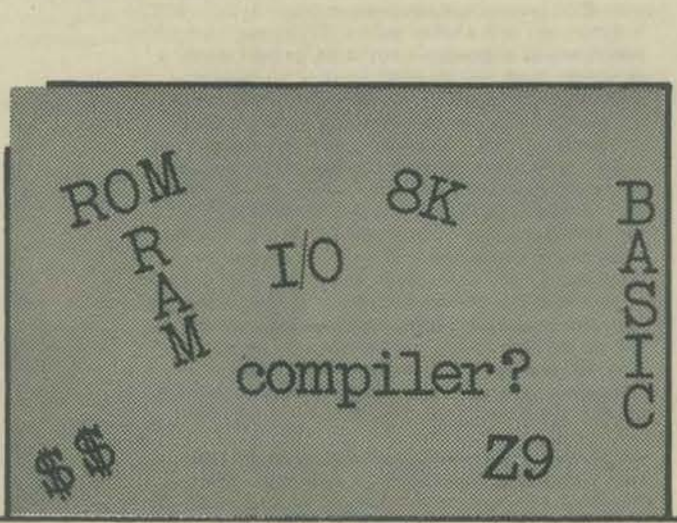
WHICH BASIC?

There is not any one standard BASIC (yet). The question is which BASIC should we choose to implement. A smaller (fewer statements, fewer features) BASIC is easier to implement and (more important) takes less space in the computer. Memory is still expensive so the smaller the better. Yet maybe we can't give up some goodies like string variables, dynamic array allocation, and so on.

There is a standard version of BASIC which is to be the minimal language which can be called BASIC. It's a pretty big language with lots of goodies. Maybe too big. Is there any advantage to being compatible with, say, the EDU BASICS? We don't have to make any decision yet; but the time will come . . .

COMPILER OR INTERPRETER?

We favor using an interpreter. An interpreter is a program which will execute the BASIC program from its textual representation. The program you write is the one which gets executed. A compiler converts the BASIC program into the machine code for the machine it is to run on. Compiled code is a lot faster, but requires more space and some kind of mass storage device (tape or disk). Interpretative BASIC is the most common on small machines.



HOW MUCH MEMORY? AND . . . WHAT KIND?

Can we make some guesses about how big the BASIC system will be? Only if you don't hold us to it. Suppose we want to be able to run a 50 line BASIC program. We need about 800 bytes to store the program, another 60 or so bytes for storing program values (all numeric) without leaving any space for the interpreter and its special data. Past experience has shown that something like 6 to 8 Kbytes are needed for a minimum BASIC interpreter and that at least 12K bytes are necessary for a comfortable system. That's a lot of memory, but not too much more than you need to run the assembler. A lot of BASIC could be put into ROM (Read Only Memory) once developed and checked out. ROM is a lot cheaper than RAM (Read and Write) memory, but you can't change it. It's lots better to make sure everything works first.

But . . . if we can agree on some chunks of code and get it properly checked out, some enterprising person out there might make a few thousand ROMs and save us all some $$$. Let's see now . . . how about ROMs for floating point arithmetic, integer arithmetic, Teletype I/O . . .

DATA STRUCTURES

Data structures are places to put things so you can find them or use them later. BASIC has at least three important ones: a symbol table which looks up a program name, A or Z9 or A$, with its value. If we had a big computer where space was not a huge problem, we could simply preallocate all storage since BASIC provides for only 312 different names excluding arrays. When memory is so costly this doesn't make much sense. Somewhere, also, we've got to store the names which BASIC is going to need to know, names like LET and GO TO and IF. This table gets pretty big when there are lots of statements.

Lastly, we need some information about what is a legal BASIC statement and which error to report when it isn't. These tables are called parsing tables since they control the decomposition of the program into its component parts.

SHOULD OUR BASIC BE TIME SHARING?

We don't think we'll try to time share our 8080 among many users. While there is no fundamental difficulty in making a multiuser system, it doesn't seem worth the effort.

STRATEGY

Divide and Conquer is the programmers maxim. BASIC will consist of a lot of smaller pieces which communicate with each other. These pieces themselves consist of smaller pieces which themselves consist of smaller pieces, and so forth down to the actual code. A large problem is made manageable by cutting it into pieces.

What are the pieces, the building blocks of BASIC? We see a bunch of them:

*a supervisor which determines what is to be done next. It receives control when BASIC is loaded.
*a program and line editor. This program collects lines as they are entered from the keyboard and puts them into a part of computer memory for later use.
*a line executor routine which executes a single BASIC statement, whatever that is.
*a line sequence which determines which line is to be executed next.
*a floating point package to provide floating point on a machine without the hardware.
*terminal I/O handler to input and output information from the Teletype and provide simple editing (backspace and line deletion).
*a function package to provide all the BASIC functions (RND, INT, TAB, etc.)
*an error handling routine (part of the supervisor).
*a memory management program which provides dynamic allocation data objects.

These are the major ones. As we get futher into the system we'll begin to see others and we'll begin to be able to more fully define the function of each of these modules.

YOU MEAN IT DOESN'T SPEAK BASIC
OR
(IDIOT SAVANT CAN'T SPEAK UNLESS SPOKEN TO)

Many enthusiasts for the INTEL series of CPU 'chips' have had the heart-break of opening up the package, and not finding a cord with which to plug it in. Now that the ALTAIR has made its debut, the next stumbling block, software, rears its ugly head. Skip Phile, the harried computer fiend, had the following comments on programming.

"It's obvious that toggling machine-code programs from the front-panel has its limitations," Skip pronounced, looking up from his Hazeltine. "First there's the matter of a 'scope driver, followed by a syntax analyzer, formatter for output, interrupt routines, and error checking logic. At the rate of one word toggled into memory every ten seconds, a good worker can have a running program by Christmas '76." Mr. Phile handed me a Government Document (reprinted here for your convenience)*. "It says here," he continued, "that the US Gov has a hardwired BASIC program that'll run on the 8008. That's something even the good old 1401 never had."

"Armed with the ALTAIR, a TV typewriter", and a good knowledge of BASIC, anyone should be able to get a low-grade STARTREK game going in about a month."

Which is good news for people who are still trying implement a toaster controller or whose callouses from toggling front-panel switches are interfering with their flute-playing.

Chuck Polisher
P. O. Box 4214
Boulder Colorado 80302

*NTIS, U.S DEPARTMENT OF COMMERCE, 5285 PORT ROYAL RD., SPRINGFIELD VA 22151
Ask about Document WGA-62-74-45

TINY BASIC

Pretend you are 7 years old and don't care much about floating point arithmetic (what's that?), logarithms, sines, matrix inversion, nuclear reactor calculations and stuff like that.

And . . . your home computer is kinda small, not too much memory. Maybe its a MARK-8 or an ALTAIR 8800 with less than 4K bytes and a TV typewriter for input and output.

You would like to use it for homework, math recreations and games like NUMBER, STARS, TRAP, HURKLE, SNARK, BAGELS, . . .

Consider then, TINY BASIC

- Integer arithmetic only – 8 bits? 16 bits?
- 26 variables: A, B, C, D, . . ., Z
- The RND function – of course!
- Seven BASIC statement types
  - INPUT
  - PRINT
  - LET
  - GO TO
  - IF
  - GOSUB
  - RETURN
- Strings? OK in PRINT statements, not OK otherwise.

Keep tuned in. More TINY BASIC next time, including some GAMES written in TINY BASIC.

WANTED – FEEDBACK! Your thoughts ideas, etc. about TINY BASIC urgently requested by the PCC Dragon.

R75-20—Weaver, A. C., M. H. Tindall, and R. L. Danielson, "A Basic Language Interpreter for the Intel 8008 Microprocessor" (52 pp., Report No. UIUCDCS-R-74-658, University of Illinois, Urbana, Illinois).

A BASIC language interpreter has been designed for use in a microprocessor environment. This report discussed the development of 1) an elaborate text editor and 2) a table driven interpreter. The entire system, including text editor, interpreter, user test buffer, and full floating point arithmetic routines fits in 16K words.

Aha! Help is available. We found this on page 68 of the January, 1975 issue of *Computer*. (Published by IEEE Computer Society, 5855 Naples Plaza, Suite 301, Long Beach Ca. 90803

NEWS FROM A SMALL PLANET

From a letter written to Rusty Whitney at OMSI –

Dear Rusty –

Many thanks for the nice letter in response to my inquiry about computing equipment for home use. You were so helpful that I'm going to impose on you for some more advice, this time about software.

After looking around at prices for used computers I've about decided that almost everything costs too much. American Used Computers is selling PDP-8I's. with 4K for about $2400, which seems to me to be grossly overpriced, and their other brokerage houses are about the same. In addition, it seems clear that microprocessor-based machines-which are certainly OK for what I want to do – are about to blossom in a big way for relatively little money. You may have seen the article in January's *Popular Electronics* in which a complete kit based on Intel's 8080 chip is being offered for under $400 with essentially no memory, and an extra $260 for every 4K of very fast RAM up to 64K (all directly addressable). The producer of the kit is also offering cheap discs, terminals, all kinds of I/O peripherals and a whole range of other equipment, for a complete system. I'm sure that other companies will be offering similar things and that this is probably the way to go for people like myself.

The only problem I can see with this approach is that so far none of the chips have software of any real sort. Most of the microprocessors are apparently intended for dedicated process control applications, and consequently are supported primarily by cross-assemblers and high-level languages requiring other machines, which defeats the purpose of a home system. Even those units that have resident assemblers won't really do for general home use, because the TV displays that almost have to be used because of cost will only display a few lines at a time and don't offer hard copy; this is likely to make debugging very difficult even for little programs. Obviously what's needed is a BASIC-type package and those don't exist.

A way out of this would appear to be offered by the fact that some chip makers are beginning to develop microprocessors that emulate existing systems for which lots of software is available. Thus single-or multiple-chip processors emulating (is that the right word?) the PDP-8E and PDP-11/05 are about to come on the market. The question is, if amateurs build machines based on these chips can they get DEC's (or whoever's) software without paying a fortune for it? I don't know anything about whether software is proprietary, whether someone who already has a DEC machine can make copies of, say the BASIC code and give it away, or what? Any comments you can pass along on this area of software for home systems would be much appreciated. I don't know quite how to pose the question to DEC or Data General — "please give me your expensive software so I can use it on my machine?"

You asked about my experience building a TV-set-based display terminal. I built one from the plans in the *Radio-Electronics* article by Don Lancaster about a year ago. It does exactly what you suggest you'd like to have – generates characters as an RF signal that goes directly to a TV set's antenna terminals. It will put up to 32 characters per line on up to 16 lines on the screen, and has a cursor. This would seem to be enough for most applications, and getting a denser display apparently requires modifying the TV set for greater bandwidth. A few comments on this design:

1. Nobody should try to build it without either substantial electronics background and available test equipment, or free access to someone who can provide this expertise. Otherwise you couldn't get it debugged, especially if you use surplus IC's.

2 It wasn't terribly cheap – over $100 with one page of memory, excluding the TV set, and at that I had a lot of the conventional parts like resistors, etc.

3. The parts were hard to get – took me over 4 months – and the IC's often didn't work right, requiring frustrating debugging and replacement.

4. The cursor can't be manipulated by software and the display doesn't scroll. I think these things are do-able with add-on circuitry, but not by me.

5. The use of a TV set, with its raster scan, puts limits on the use of the display for non-alphanumeric applications like drawing pictures, unless you have almost unlimited memory.

Southwest Technical Products in El Paso is now marketing a version of this display that seems a lot better than mine – it uses static RAMS (cheaper and simpler) has a two-page memory, and allows you to plug in cards for serial UART or parallel I/O, manual or automatic cursor control, etc., which they also make. The kit sells for about $200 without the special features and without a keyboard. I wouldn't be surprised if this cost couldn't be brought down below $100 with volume production and with some chips specifically engineered for the application. By the way, SWTP also offers a keyboard kit with ASCII encoder for about $40 – I built it and think it's way overpriced for what you get – surplus is just as cheap and a lot better mechnaically.

If you're interested in home-brew stuff you should write to Hal Singer at the Lompoc HS Computer Center, Constellation Way, Lompoc, Ca. and get on his Mark-8 User's Group Newsletter mailing list (no charge, but send some SASE's and a couple of buck to help him out). This group was started after the *Radio Electronics* article on the 8008 based computer, and now has about 300 members nationwide with very diverse backgrounds. He puts out a mimeographed newsletter once every couple of months or so with tips for construction, suggested applications, who's doing what in amateur computers, suggested circuits, etc. I think he may have the start of something big including maybe a way for amateurs to buy things at OEM prices.

Jim McCord
330 Vereda Leyenda
Goleta, Ca. 93017

P.S. Have you seen the article by Joseph Weisbecker of RCA on home computing in the August issue of *Computers*? Very interesting. He used an audio cassette for bulk storage of data and audio messages; his computer (based on the RCA COSMAC microprocessor) talks to you! Great for young kids who can't read yet. He also had some clever ideas for application.

Dear PCC –

I presently have a Radio Electronics TV typewriter and am building the Mark-8 mini computer. The TVT parts, including PC boards, power supply, connectors, and all electronics cost me $133.00. An ASCII keyboard kit which I purchased from Southwest Technical Products cost $39.00. The February '75 issue of Radio Electronics discribes a UART which plugs into the TVT board. With this UART I will be able to talk to and receive from a Teletype, record TTY messages on a cassette, play back cassettes to the TTY.

Total Cost

TV Typewriter	*133.00*
ASCII Keyboard	*39.00*
UART	*35.00*
Mark-8 with 1K words	*278.00*
Cassette tape recorder (mass memory)	*free*
Television display	*free*
	$485.00

1K static RAM'S (2102 or 2602) now available for 8 for $65.00. 4K memory PC boards for the 1K static RAM.S will be available soon for under $10.00 each. Estimated cost of 4K increases of memory – $280.00.

Now if I could just find plans for an RS-232 interface . . .

All of my suppliers have either advertised in Radio-Electronics or been mentioned in "Letters to the Editor", since the TV typewriter article. The Maxi-kit with 1K memory (1-8008, all resistors, all .1 disc capacitors, and all 7400's, plus 8-2102's) for $145 is sold by M & R Enterprises, P. O. Box 1011, Sunnyvale, Ca. 94088.

Bill Godbout Electronics, Box 2673, Oakland, Airport, Ca. 94614, Advertising, in Feb. 75 issue, page 98 lists 8008's for $50.00. A kit with the 8008 8-1K static memories, and 1-G25203 (8 x 256) EROM for $130.00.

Also in Feb. issue, page 27, a new TV typewriter. Full kit less keyboard, power supply and case selling for $175.00. (Careful shopping around would get all the parts including the power supply for $130 to $140). Also available with new TVT II are optional pluging. UART board, parallel interface board, cursor control board, and off-line edit and screen board.

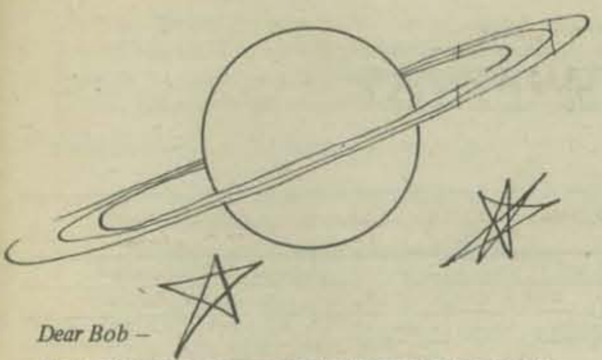
James Fry
4249 N. Lockwood
Toledo, Ohio 43612

home computers

Mr. Albrecht –

There can be no possible dispute about the microcomputers being the thing of the future. Just look at all the hobbyists that the recent articles have stirred up. Within a year the number should more double. Costs for parts are constantly dropping, even with inflation.

Gary T. Post
475 Brooks
Lemoore, Ca. 93245



Dear Bob –

Some of your readers might be interested in knowing that there are groups of amateur radio operators in the U.S. and throughout the world that listen in on transmissions from orbiting earth satellites.

The first group of these "space eavesdroppers" was formed in the early '60's in the Kettering School in England. The organization in the U.S. is the North American Branch of the Kettering Space Eavesdroppers. The main event everyone is waiting for is the joint Appollo-Soyuz mission in the summer of 1975.

No amateur radio license is needed but you do need some equipment for receiving. If you are interested and have access to a computer I'm sure this organization would be grateful if you would like to help compute satellite orbits, and distribute them to other listeners.

For more background information, radio frequencies and sample satellite tracks you can contact me.

By the way, Bob, could you give me some information on the "enterprizing group" on the East coast that is planning a PCC type center?

Charles Shapiro
40 Acacia Ave.
Chestnut Hill, Ma. 02167

from Manchester & Hertfordshire

Dear Bob,

The use of mini computers for education in this country continues to grow, particularly in the teacher training sector. As you will know, users of these systems have formed as association called MUSE (Mini computer Users in Secondary Education). They have been active now for about 2½ years and shortly are to hold their third annual working weekend. One of their members tells me that his school system is shortly to be upgraded from an EDUSYSTEM 20 to include dual DECTAPES and the capability for OS/8 BASIC. Perhaps you could suggest some educational users of similar systems in the States who could usefully exchange ideas.

I have enclosed further issues of the U.K. Newsletter 'Computer Education News' and also details of an attempt I've made at writing about the technology associated with computer-based learning. Best wishes.

J.J. Turnbull
Head of Education Applications Sector
The National Computing Centre Ltd.
Quay House, Quay Street
Manchester M3 3HU

Dear People's Computer Company,

Over the last six months occasional copies of *People's Computer Company* Newsletter have been falling into my hands and I have never failed to be invigorated and excited by it. We are an offical organization within England responsible for promoting good computer education within the county of Hertfordshire. We make use of a DEC System 10 computer situated at the Hatfield Polytechnic which is about twenty miles to the north of London. We provide educational services for all the schools and colleges within Hertfordshire who make use of this system and are prepared to talk to anybody about computer education.

We very much want to carry on receiving copies of this Newsletter and we have a suggestion which would make our administrative burden rather simpler and which would probably be amenable to you. As part of our activities we publish a regular flow of Bulletins and of various paper-backed texts – programming manuals, package diets, and so forth. Would you be prepared to barter these for continuing copies of the Newsletter? I am enclosing a copy of our latest Bulletin and a few other of the lightweight materials with this letter so that you can see what we produce and at the same time I am despatching you by surface mail the rather heavier material. An up-to-date list of what we publish is usually printed in the back of our Bulletins.

If you are interested in the exchange of programs then that could easily be arranged too, although most of our programs are written in DEC BASIC which is rather different from Hewlett-Packard BASIC particularly over text handling facilities.

J. Jaworski
Advisory Unit for Computer Based Ed.
19 St. Albans Road
Hatfield, Hertfordshire.
AL10 OHU

machine language vs assembler — an opinion

Dear Bob –

I just realized that you are in the planning stages of an assembly language book. Please do not think that the following is to put down your project; I am sure that whatever you do will be well received. For some time now, I have had some radical ideas about assembly language? if you have opposing views, I hope that you may find time to write, that we may reason together in good humor.

It is my position that assembly language, being the first level of software languages, is by far the most inefficient lagnuage, including machine language. Its use should therefore be discouraged (in favor of octal or hexadecimal look-up charts); my reasons are:

1. In general, an Assembler is a software system for converting three letter alphabet codes into machine code on a one-for-three byte basis. This directly implies that a machine language program may run in available memory, but an assembler version will not fit, and thus, means that all software modification (and there is always modification) must be done outside the computer, perhaps on paper-tape TTY equipment. Not only is this slow, but it automatically excludes the editing advantages of software systems internal to the computer.

2. In actuality, the situation is worse than described above. In general, a Symbolic Assembler reserves several alpha-numeric spaces (bytes) in each command for symbolic addressing and carriage-return (in the usual TTY form). This means that a typical assembler program may be eight or more times the bit-length of the equivalent machine code.

Symbolic addressing means that you can specify a "jump to 'HOP' " command, where "HOP" has labeled a statement somewhere in the program; this is quite helpful in program modification. When working in machine language, each "jump to" includes the actual address of the instruction you wish to "jump to." If additional steps are inserted in the machine code, not only must all succeeding steps be "bubbled up," but all jump commands referring to addresses in the modified area must also be altered. A Symbolic Assembler will generate tables to do this automatically during re-assembly after each modification.

But a relatively minor machine language program (similar to a conventional re-locatable loader) can be written to do the same thing; "jump to" statements can be identified and, if the jump is to a modified address, the jump address is changed. The operator can step through a program on a (machine operated) CRT display, and on command, cause a software system to insert a NO-OP, bubble the program(s) up, and change internal "jump to" addresses. Thus, a relatively minor machine language program allows on-line program editing while a major Symbolic Assembler may allow only off-line modification.

3. The final advantage of an Assembler is the use of mnemonic codes, which seem to have a greater affinity for human memory than octal numerics. After several months of intense 8008 programming, I am just now beginning to reliably remember the numeric machine codes. But programming is not a "sit down at the terminal and start in" job; rather, it is a pencil-and-paper process in which logical processes are ordered, erased, and squinched in-between others until the desired effect is predicted. It is thus not hard to use a simple octal conversion list for the appropriate commands once the concepts are firm.

The Assembler version of a program, being outside the computer, can be printed out easily, but must be stored and maintained in the assembler version, in case future use might show the need for modification. The machine-language version must be masked, translated, and sent (to TTY or CRT) for examination. But during this process, a relatively minor program (compared to the Assembler system itself) could also output assembler alphabet codes. A larger system could output an English interpretation in words, instead of mnemonic codes. In any case, machine language coding can do what assembler coding can do, far more efficiently.

The above arguments are critical in a small-memory system. With a disk (or semiconductor disk replacement) you can better afford to be sloppy. But, for a while at least, a whole lot of people can't really afford it.

It is easier, not harder, to program in machine code, but lots of people are afraid of it. Non computer-type scientists ridicule it. Even computer people are far more concerned with higher-level languages. I myself believe that small companies will presently start offering programmed ROM's for personal computers; a pre-programmed APL system would really tear things open. Such a system exists for the 8008, I believe, in Micro Computer Machines MCM/70. I have seen this desk-top computer operate, and it is really a powerful unit; but the price is creeping up over 5K now, so it is not what it should be. Generously, there may be $1000 worth of electronics in the thing; the money goes for the APL system. But we see the future, and there, but for $5K would go I.

Terry F. Ritter
Vice President
Dantco
2524B Glen Springs Way
Austin, TX 78741

HOW DO DRAGONS OPERATE TTYS?

For The Dragon's Lair at the PCC –

SO HOW DO DRAGON'S OPERATE TTY'S? I mean those claws aren't too advantageous, even for electric typewriters. Unless you operate on a voice instruction system, like Hal in 2001, A Space Odessey. And Hal can even have conversations! Simple, right? Wrong! An article in Pschology Today tells why. The article from a May 1974 PT is called "Why Computer Don't Talk Like Hal in 2001." The thing that attracted it to me was a pair of store bought teeth reading a paper tape.

After reading the article, you find how very very complex OUR language is and how simple BASIC is (or for that matter any computer language). It retells an article about hte Psychiatrist program at MIT called "Eliza" and it's difficulties. Then it shows you how one program analyzes a single English sentence. When you finish the article you find that it is a long, long way off before computers can talk English like Hal.

During my vacation, I visited the OMSI Computer Room. I was impressed. Here, at the Medford Mid High, students know only programming, logging and running only BASIC, along with very elementary Machine. There at OMSI, students (same age) know advanced programming techniques, how to alter computer languages, how to pick new computer hardware, know exactly how the computer works. Not only are their students advanced, but so is their hardware; DEC's SUPER BASIC using EDU/CRUNCH, along with it's accompanying machine language. Many many TTY's and 3 screens, for games and computer art. Compare that to our 2 Mid High TTY's connected by phone to a crummy HP2000F! If everybody is like that in the Computer World (all I know about this world is from the PCC and what's in Medford), we have got a long way to go! (baby).

Though some space is there at the OMSI, it seems that every nook and cranny is used for a TTY, screen or computer (and their manuals . . . whew!) But there is always space for a newcomer who didn't even make pre-arrangements, ME. I played a couple of games, asked questions and looked around. There was a lot to see in that small room. And. I thank the OMSI bunch for letting me see it.

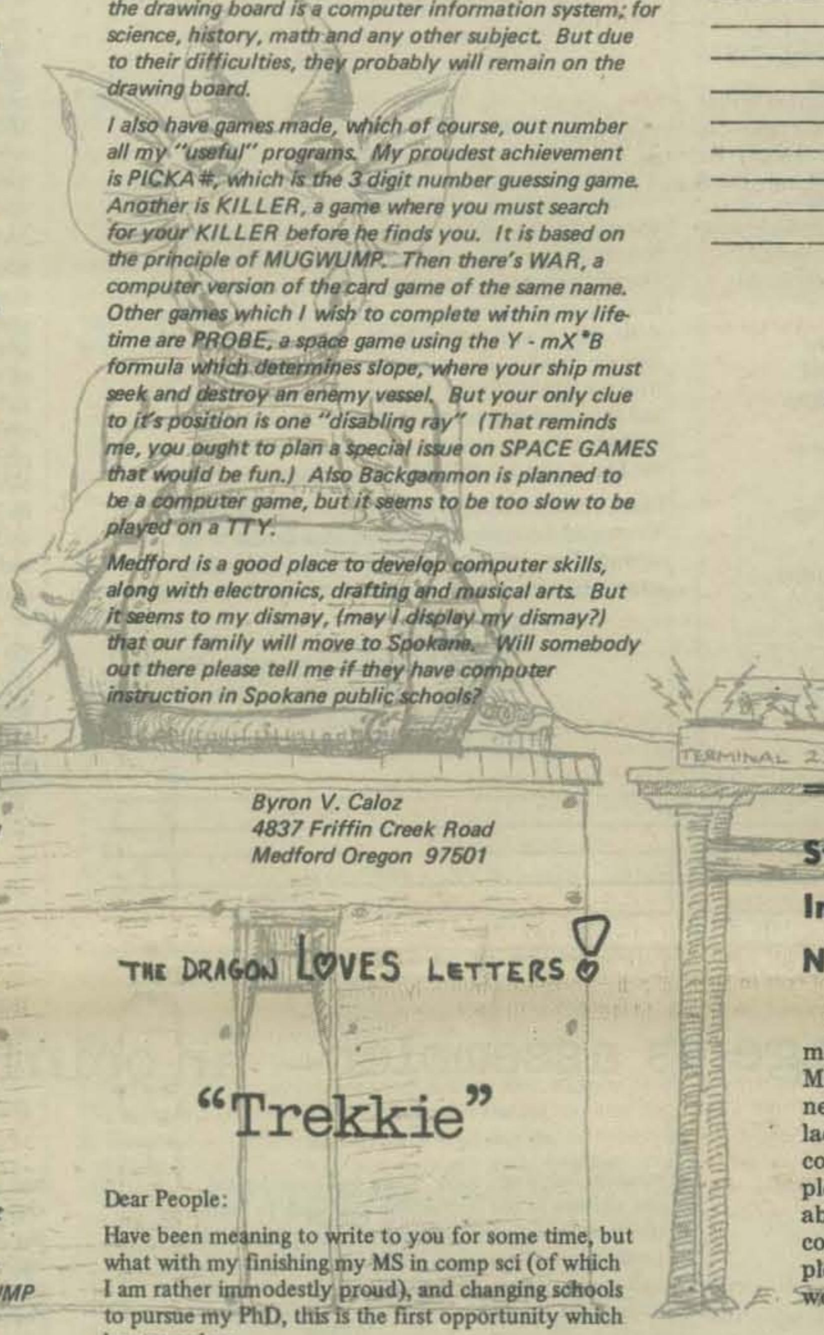
Now I got to brag about myself. Not actually brag, but to tell you guys why I take your paper or magazine. . . and what it does for me. Lately, with techniques from DAVE OF THE CAVES, TAXMAN, SNARK, MUGWUMP The Programmers Toolbox and anything else, I have splurged on writing programs. Though some have not gotten on tape yet, I've made numerous dry runs to check for mistakes. (One of our computer instructors prefers me to write "educational" program instead of games. "Games", he implies, "are OK to write, but just as long as they are good, and you don't waste paper or time.")

Here are some of my very few Public Service Programs:

Dinner – a program for planning dinners. This uses the BRK function of 2000F and makes the runner of the program put all 4 basic foods into the dinner, while computing times for preparation, at a given serving time.

Number Line – a program which takes over the tedious job of making number lines. It draws up to 10 number lines on a page, and prints the assignment data from input. I'm sure many a math teacher would like a neat copy of number lines.

SUMWRD – this explores the use of basic English laws into making words (I've found 16 different types of syllables possible in English words).



I am also working on a program which explores the human approach to understanding langauge. Also on the drawing board is a computer information system; for science, history, math and any other subject. But due to their difficulties, they probably will remain on the drawing board.

*I also have games made, which of course, out number all my "useful" programs. My proudest achievement is PICKA#, which is the 3 digit number guessing game. Another is KILLER, a game where you must search for your KILLER before he finds you. It is based on the principle of MUGWUMP. Then there's WAR, a computer version of the card game of the same name. Other games which I wish to complete within my lifetime are PROBE, a space game using the Y - mX *B formula which determines slope, where your ship must seek and destroy an enemy vessel. But your only clue to it's position is one "disabling ray" (That reminds me, you ought to plan a special issue on SPACE GAMES that would be fun.) Also Backgammon is planned to be a computer game, but it seems to be too slow to be played on a TTY.*

Medford is a good place to develop computer skills, along with electronics, drafting and musical arts. But it seems to my dismay, (may I display my dismay?) that our family will move to Spokane. Will somebody out there please tell me if they have computer instruction in Spokane public schools?

Byron V. Caloz
4837 Friffin Creek Road
Medford Oregon 97501

THE DRAGON LOVES LETTERS

"Trekkie"

Dear People:

Have been meaning to write to you for some time, but what with my finishing my MS in comp sci (of which I am rather immodestly proud), and changing schools to pursue my PhD, this is the first opportunity which has turned up.

I have greatly enjoyed reading the back issues of the PCC and am delighted to find a group which is dedicated to educational and recreational uses of computers. It is encouraging to note that 'fun' uses of computer time are becoming somewhat more legit and no longer means instant expulsion from the computer lab if the director catches you!

Besides being a computer phreak, I am a "Trekkie." Nothing fanatic, you understand, (i.e., no pointed sideburns [or for that matter, pointed ears]), but nonetheless still enough to go around attaching 'Star Trek LIVES. bumper stickers to things. Could you please send a listing of Star Trader? (For ten dollars, I can do a lot of typing) I am also very interested in exchanging listings with any of your readers who have versions of Star Trek or Spacewar, or the like. Here there are two BASIC verions: one gives you a script-type output, the other is the usual 4094 sectors with 64 quadratns galaxy type. There is also an APL version, complete with super-novas and gravitational anomalies. I'll be glad to make these available to anyone who wants to trade. Also of particular interest are any 'graphics' versions, such as for the DEC GT-40 system. I realize that this may precipitate a flood of mail as that is the usual consequence when anyone mentions "Star Trek" in print, but as they say, KEEP ON TREKKIN'!

Bob Wier
1104 S. College Rd. No. 3
Lafayette, Louisiana 70501

announcement

MINNESOTA EDUCATIONAL COMPUTING CONSORTIUM

MECC

1925 Sather
(Hwy. 280 at Roselawn)
Lauderdale, Mn. 55113
Telephone (612) 296-2789

Statewide Instructional Computing Network Is Up

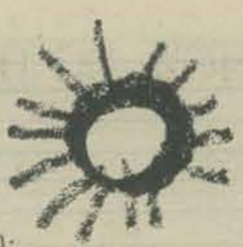
Minnesota education reached a milestone in October 1974, when MECC's instructional computing network became operational. Still lacking some telecommunications connections from schools to multiplexor stations, the network is about 90% complete. Even so, the completion of the thirteenth multiplexor station at Duluth was noteworthy.

The instructional computing network, using the thirteen multiplexors and a wide variety of telephone company equipment, connects the 800 plus terminals to the five different computer locations. Approximately five hundred terminals are in the Twin City area and thus use the dial-up network for telecommunications.

The telecommunications installation—a significant undertaking by anyone's standards, has been accomplished because of the efforts of many. Mr. Richard Williams, of the Bell Telephone Company, has been an invaluable member of the MECC telecommunications project. Mr. Larry Bueskgans of Loonam & Associates has done an outstanding job of coordinating and actually installing all of the multiplexor equipment.

The telecommunications network is a tribute to the cooperative efforts of educators in the field and their patience.

UNIX HORS D'ORUVES



UNIX is a general-purpose, multi-user, interactive operating system for the Digital Equipment Corp. PDP-11/40, 11/45 and 11/70 computers, developed by Ken Thompson and Dennis Ritchie at Bell Telephone Laboratories, Murray Hill, New Jersey. It qualifies as far and away the most comprehensive, general, and sophisticated operating system available for any minicomputer today.

Some of the important features of UNIX are:

1. A hierarchical (tree-like) file system structure that permits removable sub-structures;
2. Fully compatible I/O to and from files, devices, and other concurrently-running jobs (or processes);
3. The ability to initiate asynchronous processes;
4. System command software selectable on a per-user basis (i.e., each user may have a different command language available into which he is dropped at login time);
5. Over 100 "subsystems" including a dozen languages.

UNIX was developed on the PDP-11 in 1971. Limited earlier editions existed on the PDP7 and PDP9 in 1969–70. There are now more than 70 UNIX installations within and outside the Bell System. Some representative non-Bell users include Columbia University; the Oregon Museum of Science and Industry; Harvard University; and the Boston Children's Museum, the first licensed non-Bell users.

Some of the more important and interesting programs available under UNIX are:

an assembler that vaguely resembles PAL-11R;
a versatile text editor based on QED;
a compiler for a language resembling BCPL with the addition of data types and structures, called C (and successor to, of course, B);
a pseudo-BASIC interpreter;
a text-formatting program in the mold of RUNOFF;
a FORTRAN compiler;
a SNOBOL interpreter;
an ASCLI sort-merge program;
a symbolic (albeit non-interactive)debugger;
a form-letter generator;
a very flexible standard user command interface called the Shell;
a driver for the Federal Screw Works VOTRAX voice synthesizer;
the games of Wumpus and Chess, plus a "learning" version of tic-tac-toe;
and an assortment of programs for utility and maintenance purposes (list contents of director, check file system integrity, copy files, etc.)

I should be noted that UNIX is fully self-supporting, largely self-documenting, and extremely reliable. Delivery of the system includes source code for the operating system itself and all programs. Over 90% of the system is written in C; once you become acclimated to it (which took me a month of very sporadic poking around; a more reasonable effort could be accomplished by an experienced programmer within a few days), C is a very easy language to use.

We brought up our initial shipment of UNIX in minutes without a hitch. We run UNIX on a PDP-11/40 computer with 48K words of core memory, two 1.2-million-word disk cartirdge drives, a line printer, nine terminals including a Teletype, a DECwriter and six alphanumeric CRT's, a QUME high-quality 30-cps character printer, a voice synthesizer and (soon) a general-purpose relay/switch interface box. A useful UNIX requires a minimum of 32K words of core in all, plus a disk, hardware multiply/divide, memory management, a real-time clock, and, of course, an 11/40, 11/45 or 11/70 CPU and one or more terminals.

UNIX is extremely reliable. We have experienced no software-related crashes since receiving the most recent edition of UNIX about three months ago. As this is being written, our system has been up for four consecutive 24-hour days without a crash; the most recent crash before that was due to my own human error. In fact, the system has been so reliable that I was lulled into complacency and recently lost 130,000 characters of information when our second disk drive went flakey — I had not made a backup copy of that cartridge for a coupleof months. In no case has a processor-related hardware crash caused any loss of data.

The central philosophy underlying the development of UNIX was to provide a reasonably efficient environment for programming. The features of the system certainly realize this goal: the file structure, available languages, and process-orientation of the system are all very convenient. Further, the system's protection system and flexibility suit it well to the requirements of a high school that has advanced computing courses or a university, and to the off-the-wall public interaction we experience here at the Children's Museum. We have been sharing development work on the system with Harvard University, which has an 11/45 UNIX that often supports 20 users, since it was first delivered to us. This co-operation has led to the implementation of a number of games, utility programs, and complete sub-systems like an extended BASIC (similar to Digital's standard BASIC/RT-11). Currently under development are a general-purpose information storage and retrieval system, dubbed *The Information System* †, and an extended and slightly more flexible shell. A high school student is implementing FOCAL, which should be ready within a few weeks.

UNIX is available for a nominal distribution fee (at last I heard, $150) to users who sign a licensing agreement that requires that the system be used solely for educational purposes. Users who are not willing or able to sign the agreement may purchase a license at a cost of $20,000 (again as of my most recent contact with the folks at Bell.)

The system is not covered by any software support clause. Hence, it is not well suited to the needs of the small school, or indeed of any school that does not have a professional person with quite a bit of programming experience; although the software is, as I pointed out, extremely reliable, one still needs to be able to understand what's going on in its innards to be able to diagnose hardware problems that crop up spontaneously in defiance of DEC diagnostics, and one needs to become fluent in C to write new software and make modifications to existing code. Nonetheless, UNIX is an outstanding bargain for installations that can make use of it.

Much of the information in this article has been excerpted and paraphrased from a document entitled *The UNIX Time-Sharing System* prepared by Thompson and Ritchie. Readers who would like copies of this document, or other information on UNIX may contact me directly; where appropriate, requests for licensing information, etc. will be forwarded to Bell, but I will attempt to answer technical questions so as to minimize the strain on the UNIX development people. My address is:

Bill Mayhew
Director, Computer Systems Development
The Children's Museum
Jamaicaway
Boston, MA 02130

† Copyright © 1975 by the Children's Museum

Dear Bob,

Good Heavens. Where to begin?

I received yesterday the two issues of PCC that have been published thus far this year thanks to your intervention (I presume) with your crackerjack subscription department. My thanks, I think; sometimes PCC simply frustrates me because it's packed with too many far-out ideas and I don't have enough time to deal with all of them and still get my work done.

I also realized that by now I must owe you about an entire issue's worth of articles, both on what we're doing and on UNIX and on our new center. I promise I'm going to get right on them. Tonight I will write about UNIX; tomorrow about our new configuration and what we're doing with it.

Things here have now reached the point where I can't even keep track of them. I have two dynamite high school kids, Mike Fitzgerald and Eric Feigenson, working for me doing software and even a little hardware development. Mike is a senior at Copley HS here, and works at the Museum with me for about an hour and a half a day for credit under the school's flexible-campus-like program. He wants to be a computer "techie" when he "grows up" (heaven forbid) and is looking for some inspirational ways of getting training that will lead to a job in the reasonable future. He has good skills, and is slowly learning all there is to learn about PDP-11's and VT05 terminals, but our system is finally becoming reliable (knock on wood) and the opportunities for hands-on experience fixing it are rather small. He's also getting quite sharp with software and now uses UNIX's high-level systems-programming language, C, exclusively (BASIC is currently implemented has no matrix operations and is too slow when it needs to be swapped onto a moving-head disk, and C is much better once you get used to it).

Eric is a junior at Newton South High School where they run RSTS on a system smaller than ours (used to be there weren't any systems smaller than ours!); he had a work-study job with them during the summer and has experience as a System Manager. His software proficient in C.

Yesterday I sat down with a couple new staff people and was amazed to discover the range of games and things that we've implemented. The general public has access to seven of them: UGUESS (1 to 100 number-guessing), 10GUESS (1 to 10 number-guessing), a modified WUMPUS, HANGMAN, TICTACTOE, MATHDICE (an adding game for itty-bittys), and 31 (the game of thirty-one). We also have tried STARS and REVERSE but the level of abstraction required was too great for our uninitiated audience to deal with readily. Sitting around on the disk waiting for me to check out their instructions and interactivity are NIM, MADLIBS (which I included in my article a year and a half ago), and several others that I forget about at the moment. We also have a couple of friendly quiz things that we're going to run in conjunction with our Dentists role–playing exhibit during Dental Health Week (Month?) in February. I'm right now in the midst of rewriting the program called the SHELL which is the normal interface to the system for programmers and "real users"; the public interacts with a program called "HI" which is garbage-proof and tells the user a little about the computer, the terminal, and what things he can do with it. I'm enclosing a printed copy of what a sample terminal session might look like (annotated of course).

Next month, we're going to all-CRT terminals in our public space (thanks again to our friends at Digital Equipment). Paper costs being what they are, and DECwriters being what they are, it is the only reasonable approach.

I also recently discovered that I have a TRAVEL BUDGET that will help to pay for a maybe trip to the West Coast later this spring (although I've been saying I was going to make such a trip every spring for the last four years). Maybe will actually get there this time and will stop to see PCC.

Other than that, things are about as off-the-wall here as normal. My work load is lightening up a tad (due, I think, to the fact that I've been screaming about my overcommitted time for three years and somebody is finally listening) so I might even be able to answer some letters from readers.

Keep up the as-per-usual tuhriffic work.

Bill Mayhew
Director, Computer Systems Dev.
The Children's Museum
The Jamaica Way
Boston, MA 02130

MORE

"steal from the rich...

From our circulation department to Theodore Kramer –

"Steel from the rich, give to the poor and sell your memoirs for 30K."

From Theodore Kramer to our circulation department

Dear whoever-it-was-who-sent-me-the-quick-cure-to-my-financial-problems . . .

1. you don't spel to gud (r yu a teechur?)
2. I stealt from the rich to make the enclosed book order
3. does PCC come under your definition of poor? If so I have taken care of part 2 of your good advice.
4. I really meant $30K yearly, so I would have to sell a memoirs each year to take care of my needs.
5. What's a memoirs? If you could describe it to me and I can find one, will you buy it from me for 30K?

Now, on to new stuff.

I am interested in starting a dialog with computer professionals who are into such ideas as home computing, fun computing, learning computing, etc., and who would like to put something together which would be elegant and economical enough to attract capitalization. I have some embryonic ideas of my own which would benefit by exposure to the cold light of reason and experience.

My experience with computers is 99.44 per cent IBM oriented. I speak BAL and COBOL with equal facility, can handle FORTRAN with the help of a book, and have done a few things in GPSS. "MY" current computer is an IBM 370/168 with more core, disks, tapes scopes, bells, and whistles than anyone in their right mind could ever use. They tell me that TSO, the slow assed "time sharing" system up on this beast chomps 700+K!!! 700K!!!!! Can you imagine any program not written by imbeciles or hardware salesmen that burns that kind of core??? I won't tell how much OS, the super-sophisticated operating system eats . . . you'll never stop laughing. Imagine, this software is only to get you to the point where you can put and take from the system. Application programs tend to run about 100K each, averagewise. But it's worth it because of the fantastic response time and the many users serviced, you'd think . . . right?? WRONG??? WRONG!!! The damn dumb monster sometimes times out during log-on for crissakes.

Can you send me names/addresses/phones of people/organizations in Southern California who are into alternate people-oriented uses of computers?

What is a Digi-log??? Where can I find out how/when to get one?

Enuff. Please send me the name of whoever or whatever you are. It feels funny to write a letter to a company.

Theodore Kramer
5307 Sepulveda Blvd.
Van Nuys, California 91411

Comments

Dear PCC Dragon –

The world has turned into several bright and beautiful hues since I opened your package of goodies (Vol. 1-3). Unless I miss my guess, I see a broadening of the interests represented in the paper and for one, would like to encourage you to continue to report broadly. Personally, I am dedicated to the effective use of computers by lawyers, an extremely conservative group which is currently organized into very small and inefficient provider-groups called law firms; some of these firms can be pursuaded to improve their methods of delivering their product but the inability to organize a large group of users seems to prohibit any large scale time-shared systems. Thus, the progress being made in the use of small inexpensive computers is central to progress toward my goals.

Some comments as I flip through the pages of Vol. 3, No. 3:

P. 2-3: enormously useful information.

P. 4: a great idea but I can't do more than get the flavor of it from the description. I would love to have a listing of your parsing algorithm. The key seems to be in the statement T = C$, but I don't know what that does aside from a straightforward assignment (and if that's all, it seems redundant). So, gladden my heart and show me the "frazzle-dazzle look-up": I should think everyone could benefit from it and would encourage publications.

P. 8-9: I am scuttling my concern abou the expense involved and am ordering COMPUTER LIB. What about putting together a slightly less personal and more comprehensive primer on the order of P. 10 and 11 which is the best-by-far explanation of Modems and Acoustic Cuplers I have ever seen. By all means KEEP LEVINE WRITING !!!

P. 12-13: Fantastic; just what the world needs.

Misc: any way to collect the info on TTY and issue it in booklet form? Also, my survey of BASIC instruction books let me to prefer PROGRAMMING TIME-SHARED COMPUTERS IN BASIC by Eugene Barnett (Wiley, 1972) and I looked at and own a lot of similar books, none coming close in terms of style and content.

I've just spent about an hour trying to read the listing on P. 13 of No. 2 (Bulls and Cleots) and am utterly stumped. There are many indications that the code is not very good, e.g., Line 420 resetting P9 to the only value it could have at that moment and inside a loop no less! Yet I'm not very practiced at reading other people's programs and am not familiar with the use of push-down stacks in BASIC so will allow as how I may be the one at fault. In any case, however, I'm confident that few of your readers could read that program given the absence of any remarks or documentation of any kind, not even a variable description table. I'm amused to see a computer that can compute the value of a user-defined function before its been defined (Line 490 and 740). I say all this because I can appreciate the tremendous value to be gained in reading other people's code and want to use that technique to learn their "tricks"; but given the effort expended here and the little I learned, I'm beginning to wonder.

Keep up the fantastic and much appreciated work!!

Beardsley Ruml, II
3306 Cathedral Ave.
N.W., Washington, D.C. 20008

U of Oregon

The Department of Computer science at the University of Oregon is pleased to announce its Summer Program for 1975. A few of its key features include –

1. **Five workshops, of one week each, designed for elementary school teachers, school administrators, social science teachers, science teachers, and business teachers respectively.**
2. **A number of low prerequisite courses covering a broad variety of computer science topics. These include programming language courses in BASIC, FORTRAN, FLECS, and PL/1 and courses on Environmental Modeling and Simulation, Computer Graphics Devices, Natural Language Processing, Information Systems, and Introduction to Computers in Education.**
3. **A comprehensive set of courses leading to a Master's Degree in Computer Science Education, designed for teachers. Depending upon one's background, this requires 3-4 summers to complete. A full program of study is available at each level for people in this program.**

SUMMER SESSION

An improved program consisting of one week workshops, four week courses, eight week courses, and individualized programs of study. Designed for educators, computer science majors, and people needing computer science service courses.

4. **Courses for computer science majors at the undergraduate and graduate level. Students with 8-12 hours of computer science background, all the way up to graduate students in Computer Science, will find a broad range of courses available. If interested they should be able to build a program of study of up to 12 credit hours of appropriate coursework.**

Computer facilities available include an IBM 360/50, a PDP-10, and various mini computers.

If you have specific questions, please contact:

David Moursund
Dept. of Computer Science
University of Oregon
Eugene, Oregon 97403
686-4394

More general information about the Summer Session is available in the Summer Session Catalog.
Write to:

Summer Session
64 PLC, University of Oregon
Eugene, Oregon 97403

RENEW NOW

One more issue for good old Volume 3.
The May '75 issue ends this publication year . . . it's time to renew!

NSF Grant

HANOVER, N. H. – A grant of $31,000 has been awarded to Dartmouth College by the National Science Foundation to support development of a national standard syntax for graphical operations in the computer language BASIC, according to Professor Arthur Luehrmann, assistant director of the Kiewit Computation Center.

The project, which is funded from January 15, 1975 to January 15, 1976, will add graphical statements to BASIC. A national panel of experts from universities and industry will assist in the program.

The project actually began in early October 1974, under the development of Prof. Luehrmann, when a national panel held a conference at Dartmouth's Minary Conference Center to discuss the needs and prospects of such a language development Additional support in the amount of $7,400 has been received from the National Bureau of Standards, California Computer Products, Inc., Hewlett Packard Inc., UNIVAC and Tektronix.

According to Professor Luehrmann, "some of the most compelling examples of educational computer use are ones that involve graphics. Yet, it is very difficult to transport these programs to other institutions because there are no standards for writing graphical programs. We hope to change the situation so that BASIC programmers will have a standard way to write graphical programs that can be exchanged easily."

The language BASIC was developed at Dartmouth in the early 1960's by President John G. Kemeny and Mathematics Professor Thomas E. Kurtz, director of the Kiewit Computation Center. It has since gained acceptance as one of the world's major computer languages.

The panel working on this project is one of national stature and includes, in addition to Professor Luehrmann, Professors Stephen J. Garland and Larry R. Harris of the mathematics department at Dartmouth, and representatives from several other institutions.

The project welcomes correspondence from people who would like to influence its outcome or be kept informed of its progress.

Altair 8800

Gentlemen:

I read with great interest your articles dealing with the Altair 8800. We are rather excited about the possibility this mini-computer may have for us. Presently we are on a time-sharing system with a local business firm. However for the cost of our time-sharing we could develop the Altair 8800 into our own programs at perhaps some savings in the short run and certainly much saving in the long run.

We have written for more details about the Altair, but without trying to put you in an uncomfortable position we would like your assessment of the potential the Altair offers. Divine Heart is a small school, but we believe the computer has its palce as an educational tool, and as part of the curriculum itself. Certainly the computer will affect our lives more and more and as such we do have a responsibility to our students to prepare them to meet the world of today and tomorrow.

I would appreciate your honest assessment of the Altair and how you think it could be put to use in a small school setting. Time-sharing has its advantages in that we do not have to be experts to use it. I am especially interested in what you think novices could do with the Altair. We are fortunate in having someone on the staff with some background in computers and believe he could run our system for us.

Thank you for your past issues of PCC. I have enjoyed each issue very much. Next year we plan to offer a course in computer science. I do believe PCC will be in the hands of each student.

Rev. Thomas Cassidy, S. C. J.
Principal
Divine Heart Seminary
Donaldson, Indiana 46513

Next issue - more ALTAIR stuff

we all love computers but...

Dear PCC –

We all love computers, but . . . the cost of a complete system is still more than most schools can (or are willing) to spend. A lot of schools can afford programmable calculators though. The cost is much less, they are simpler to operate, more reliable, and cost almost nothing to maintain. They also offer some advantages for classroom use. For one, the programming, being more basic than BASIC or FORTRAN gives more of a feel as to what's really going on inside. Also, they can be programmed by an entire class at once with the use of prescored punched cards and each student can still run his own program without taking a lot of time inputing the program.

I have two programmables (A Compucorp 025 and a Monroe 1666, which are about the same) in my Math Lab at Edison Jr. High and use them to teach basic computer programming, for CAI (Monroe has a CAI kit with over 200 programs covering much of the 1st–10th grade math curriculum from counting to quadratic equations and matrix inversion), and for recreational purposes such as game playing. I have a number of game programs and will exchange programs and ideas with anyone interested. Some programs are versions of Number, Bagels, Mugwump, Moon Landing simulations, Nim and Tic-Tac-Toe. Please send ♥

With the recent introduction of handheld programmable models such as the HP55 and 65 ($400 – $800) and Novus programmables ($140 – $200) [nice but fewer memories and no conditional branching] there should be quite a few to choose from at even lower prices by the end of the year.

Keep up the good work. I wish there were a PCC here in L.A. START ONE ♥

Richard Moffie
3355 Canfield Ave. No. 10
Los Angeles, CA 90034

SIGCUE Publication

SIGCUE Announces Special Publication

The ACM Special Interest Group on Computer Uses in Education has sponsored a special 80-page publication, entitled *Topics in Instructional Computing,* devoted to teacher education in instructional uses of computing. Edited by Stuart Milner of the School of Education at Catholic University of America, *Topics* contains twelve refereed papers which discuss methodology, languages, resources (materials, Organizations, etc.), attitudes, effective learning environments and familiarizing teachers at all levels with the broad spectrum of instructjional computer uses. An introduction by Milner discusses computer education for teachers and cites an extensive bibliography.

Referees for *Topics* are Fred Bell (University of Pittsburgh), Judy Edwards (University of Iowa), Adele Goldberg (Xerox Palo Alto Research Center), Milner and Karl Zinn (university of Michigan). The twelve accepted papers (of more than 40 submitted) are grouped in three categories as follows:

Part One. University-Based In-Service and Pre-Service Programs

Training Secondary School Computer Science Educators
Joyce Statz
Computer Science Education for Students Training to be Teachers
Terry J. Frederick
Teacher Certification – A Computer Science Necessity
J.L. Poirot and G. G. Early
Computer–Augmented Teacher Training
Portia C. Elliott & Howard A. Peelle
A Summer Master's Degree Program in Computer Education
Peter Moulton and David Moursund
Computer Augmented-Learning Programs for College Instructors
Frederick H. Bell

Part Two. Approaches for Disseminating Ideas and Materials About Computers.

Promoting Computer Use in Social Studies: A Dissemination Effort
Donald Holznagel & Daniel Klassen
A Grassroots Approach to Teacher Training: OCCE
Tim G. Kelley
Project LOCAL – Bridging the Gap
Robert N. Haven

Part Three. Unique Implications of Computing for the Teaching-Learning Process

Classroom Communication Media
Adele Goldberg and Bonnie Tenenbaum
Using LOGO in Teaching
Richard Rubinstein
Some Thoughts on Computers and Greatness in Teaching
Thomas A. Dwyer

Topics in Instructional Computing has been distributed to all SIGCUE members on the roster as of December 11, 1974 (date of label preparation for mailing). Additional copies are available at $4.00 each from A. Ken Morton, Kiewit Computation Center, Dartmouth College, Hanover, NH 03755. All orders will be shipped postpaid; an additional charge of $1.00 will be assessed for any order which is not prepaid (including purchase orders).

$6.95

WHAT TO DO AFTER YOU HIT RETURN

or

P.C.C.'s First Book of Computer Games

(High Technology is Low Magick)

PUTTING THESE GAMES ON YOUR COMPUTER

All the games run in Hewlett-Packard 2000F BASIC. You can type in the games you want if you're careful and patient — many are listed in the back (we've left out the really long ones). Or, see the last section in the book for information on ordering magnetic tape or individual paper tapes.

If your computer speaks a different dialect of BASIC — Many of the programs should "almost" work as is. Strings and disk references (if any) may require modifications. (Check a copy of the HP 2000F user's manual.)

To reduce the size of the programs, you can remove REM's and the game's rules (have separate instructions available or how about having our book handy?). Happy crunching if your BASIC allows multiple statements per line.

PLAYING THESE GAMES WITHOUT A COMPUTER

A • before a game means that it can be played without a computer. Suggestions for "how" appear in the chapter introductions and game descriptions.

See page 27 for ordering information

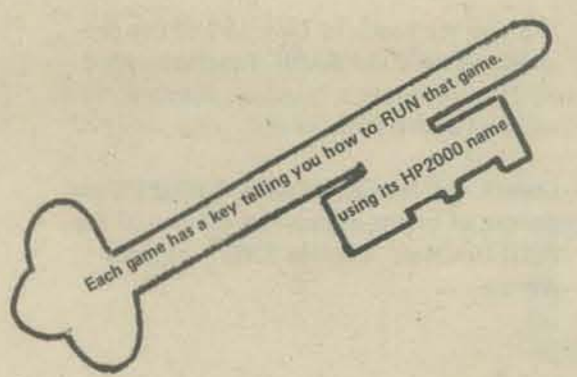


Table of Contents

Number Guessing Games

Word Games

"Nimlike" Games

Hide & Seek in 2-Dimensional Grids

Pattern Games

Board Games

Welcome to the Caves

Business & Social Science Simulations

Science Fiction Games

Last Chapter

THE MYSTERIOUS and UNPREDICTABLE RND
by Bob Albrecht

I've written a book about the BASIC function called RND, but I haven't gotten around to publishing it yet. So . . . I've decided to "serialize" parts of the book in PCC. As you follow the adventures of RND, perhaps you would send criticisms and suggestions such as "By all means, do publish this marvelous material" or (sigh), "Even a Dragon shouldn't write such bad stuff – turn those PCC pages over to someone who can write and burn all copies of your manuscript!"

RANDOM NUMBERS

Random numbers are numbers that are "chosen at random" from a given set of numbers.

Below is a sequence of random numbers. Each number is either 1 or 2.

2 2 1 1 2 1 1 1

We got the random numbers by flipping a coin. If it came up HEADS we wrote "1." If it came up TAILS we wrote "2." How many HEADS did we get? How many TAILS?

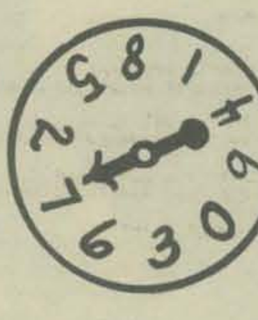
Roll a die . . . get 1 or 2 or 3 or 4 or 5 or 6. We did it 10 times:

3 1 6 2 5 1 3 3 5 6

Suppose we wanted a sequence of random numbers in which each number is 1 or 2 or 3 or 4. Easy . . . roll the die. If it comes up 1 2, 3, or 4, write it down. But if it comes up 5 or 6, don't write it down.

How can we use a die to get a sequence of random numbers in which each number is 0 or 1 or 2 or 3 or 4 or 5?

Random digits? Use a spinner. Whole numbers from 0 to 99? Use a spinner twice or use two spinners. How about a sequence of random numbers in which each number is 0 or .1 or .2 or . . . or .9? And how do we get a sequence of random numbers in which each number is 0 or .01 or .02 or .03 or . . . or .99?



Superdice! A regular die is a cube with 6 faces, numbered 1 through 6. An icosahedron has 20 faces. Two faces numbered 0, two faces numbered 1, two faces numbered 2 and so on up to 9. Here is a sequence of random digits we got by rolling our icosahedron 20 times.

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

Tell us how to use a regular die (6 faces) and a coin to get random digits?

We have a dodecahedron (12 faces) with faces numbered 1 through 12. How can we use it to get random digits?

RND AT PLAY

Below – RND is doing his random thing – simulating (imitating) a sequence of coin flips.

```
100  REM *** COIN FLIPPER
110  PRINT "HOW MANY COIN FLIPS DO YOU WANT";
120  INPUT N
130  PRINT
140  FOR K=1 TO N
150  LET R=RND(0)
160  IF R<.5 THEN 190
170  PRINT "TAILS",
180  GOTO 200
190  PRINT "HEADS",
200  NEXT K
210  PRINT
220  PRINT
230  GOTO 110
999  END
RUN

HOW MANY COIN FLIPS DO YOU WANT?20

HEADS          TAILS          HEADS          TAILS          HEADS
TAILS          TAILS          TAILS          HEADS          TAILS
TAILS          HEADS          HEADS          TAILS          HEADS
HEADS          TAILS          HEADS          HEADS          HEADS

HOW MANY COIN FLIPS DO YOU WANT?20

TAILS          TAILS          TAILS          HEADS          TAILS
HEADS          HEADS          TAILS          TAILS          TAILS
HEADS          HEADS          TAILS          HEADS          HEADS
TAILS          TAILS          TAILS          TAILS          HEADS

HOW MANY COIN FLIPS DO YOU WANT?
```

AHA! Here is RND, in Line 150.

Did you see him? In Line 150 of the program, we used the BASIC function called RND to compute a "random number" and assign it as the value of R.

Games. NUMBER, STARS, WUMPUS and dozens of others depend on the use of the RND function. Playful RND – at your service.

RND NUMBERS

The RND function generates numbers that "seem to be" chosen at random. Some people call numbers computed by the RND function *pseudo random* or *quasi-random* numbers. (Look up "pseudo" and "quasi" in your friendly local dictionary.)

We will call numbers computed by the RND function RND NUMBERS.

Instead of telling you about RND numbers, we will suggest experiments that you may try and then . . . draw your own conclusions!

First experiment – use this program to compute and print a bunch of RND numbers – as many as you like.

```
100  REM *** RND NUMBER GRINDER OUTER
110  PRINT "HOW MANY RND NUMBERS DO YOU WANT";
120  INPUT N
130  PRINT
140  FOR K=1 TO N
150  PRINT RND(0),
160  NEXT K
170  PRINT
999  END
RUN

HOW MANY RND NUMBERS DO YOU WANT?20

 .626415       .116705       .988082       .593227       .782516
 .511995       .815293       .52016        .447355       .368334
 .911773       .70078        .463951       6.59918E-02   .146018
 .946014       .848529       2.08081E-02   .723665       .895693
```

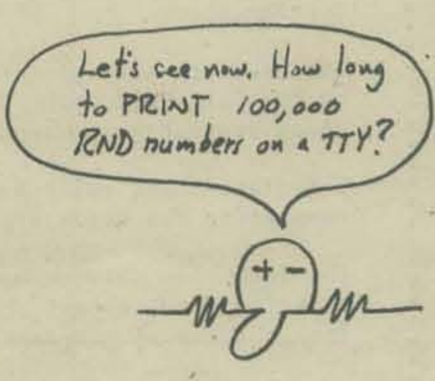
Our RUN produced a sample of 20 RND numbers. Important! If you RUN our program you will probably get a different sequence of RND numbers. (Well, we told you the RND function was unpredictable.)

OK, RUN the program and get some samples. Look at them. What do you think you know about RND numbers? Go ahead – write something down. You may be right.

HOW BIG? HOW SMALL?

How big might a RND number be? Let's investigate. Here is a program that computes, *but does not PRINT,* a sequence of RND numbers. What is PRINTed is the *largest* RND number that occurred.

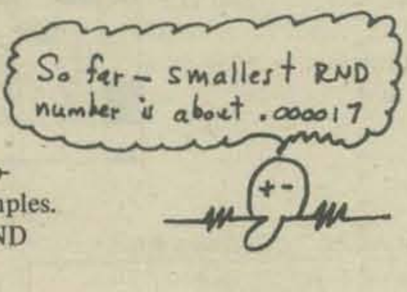
```
100  REM *** LARGEST RND NUMBER IN A SAMPLE
110  PRINT "SAMPLE SIZE";
120  INPUT N
130  LET L=RND(0)
140  FOR K=2 TO N
150  LFT R=RND(0)
160  IF L >= R THEN 180
170  LET L=R
180  NEXT K
190  PRINT "LARGEST NUMBER IN SAMPLE WAS";L
200  PRINT
210  GOTO 110
999  END
RUN

SAMPLE SIZE?1000
LARGEST NUMBER IN SAMPLE WAS .999215

SAMPLE SIZE?1000
LARGEST NUMBER IN SAMPLE WAS .997854

SAMPLE SIZE?1000
LARGEST NUMBER IN SAMPLE WAS .999908

SAMPLE SIZE?10000
LARGEST NUMBER IN SAMPLE WAS .999886

SAMPLE SIZE?100000
LARGEST NUMBER IN SAMPLE WAS .999996

SAMPLE SIZE?100000
LARGEST NUMBER IN SAMPLE WAS .999993

SAMPLE SIZE?
```


Let's see now. How long to PRINT 100,000 RND numbers on a TTY?
+ -

OK, take a chance! Say something about how large a RND number might be. This is called making a *conjecture,* or proposing an *hypothesis.* It is something you think is true, but you are not sure. You haven't proved it.

We also wrote a program to find the *smallest* RND number in a sequence of RND numbers. Here is a RUN of our program.

```
RUN

SAMPLE SIZE?100
SMALLEST NUMBER IN SAMPLE WAS 9.08585E-03

SAMPLE SIZE?1000
SMALLEST NUMBER IN SAMPLE WAS 5.70487E-04

SAMPLE SIZE?100000
SMALLEST NUMBER IN SAMPLE WAS 1.71084E-05

SAMPLE SIZE?
```



Look at our results. Write your own program. RUN your program, get some samples. Make a conjecture about how small a RND number might be.

TO BE CONTINUED

Development of the RND material was originally sponsored by Hewlett-Packard. All programs are written in BASIC for the HP2000 series of time-sharing systems. Thanks, HP for your help.

INCHWORM

. . . Karl Albrecht

Willamette High School / 1801 ECHO HOLLOW ROAD, EUGENE, OREGON 97402 PHONE (503) 689-0731

Lane County School District No. 52, Ray Klappenbach, Superintendent

People's Computer Company
PO Box 310
Menlo Park, CA 94025

Gentlemen:

Enclosed are several inch worm puzzles that were developed by the Data Processing class of Willamette High School. The class started just this semester, which began January 27, 1975.

The diagram and rules for each game and the programs for completing the games are given on each page.

The Inch worm puzzles have been used as an introduction to stimulate interest and to have the students start thinking in terms of programs, codes and problem solving techniques.

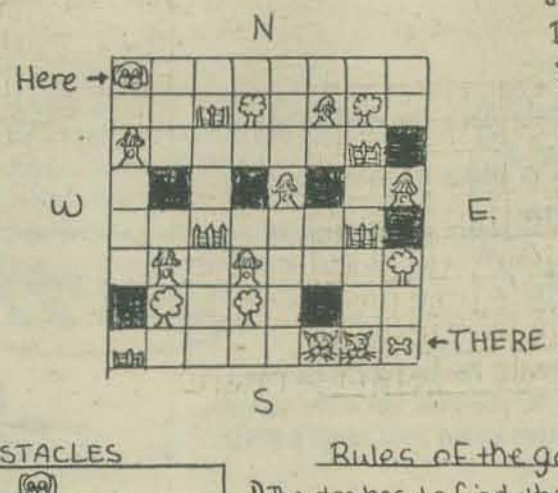
I hope that you may be able to use one or more of the puzzles or possibly could send a note back to the class. The names of the students who prepared the puzzles are attached to each puzzle. Thank you for your interest and enthusiasm.

Sincerely yours,

Paul Hoggatt

Paul Hoggatt
Data Processing Instructor

Tralina Streeter
Sherri Mord
Delores Hall
Suzy Owens



OBSTACLES

Dog –
Tree –
Fence –
Fire hydrant –
Dark Alley – ■
Cat –
Bone –

Rules of the game:

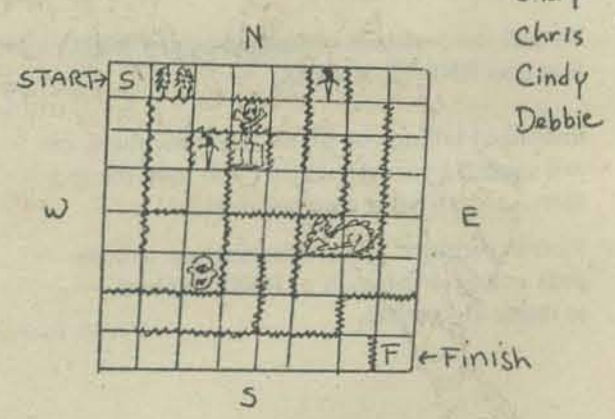
1) The dog has to find the Bone in 14 Steps starting at Here.
2) He cannot go through the Trees or Fences, he is to go around them. (THey are blockades.)
3) For each route you have to go Through 1 Fire Hydrant and 1 dark alley, but only 1 of each.
4) THe cats may be substituted for a Fire hydrant or a darkalley. (In case That you need to use a fire hydrant or dark alley). Remember, only one fire hydrant or dark alley or cat substitutes may be used!

For Example: you must use the obstacles like THIS: ♟ + ■ , 🐱 + ■ , ♟ + 🐱 , or 🐱 + 🐱 to get to the bone.

Programs: 1) E S S E E E E S S S S E E E S
2) S E S E E E S S S E E S S E
3) E E E E E S S S S S E S E S
4) S E S E E S S E E S E S S E

There are a few more routes also.

Cheryl Davis
Chris Taekker
Cindy Dickens
Debbie Heffentrager



Rules: You can go through the puzzle in ANYWAY You can find that is not blocked by a FencE or object. You must start at the Start and finish at the Finish.

Objects that could harm you:

- Evil Forest
- Skull + crossbones
- Knife (2)
- castle with red baron (he's evil)
- Dragon
- One-eyed cyclops.

Program: SSSSSENENWNNENEESSSENNENE-SSSSSWWSWNNWSSWWWSEEEEEE-NES-Finished.

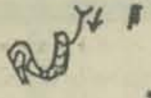
Mike Gongle
Dave Fuhrer
Lance Robinson
Marie Born

N

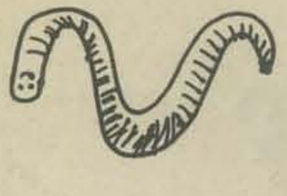
Here-	H	H	A	T	C	E	H	L	
	O	R	B	O	A	T	K	m	
	N	S	J	N	O	O	N	E	
W	L	A	E	P	F	E	A	N	E
	Y	E	L	L	A	G	R	T	
	O	G	E	A	L	L	J	E	
	R	A	G	T	L	A	I	R	
	U	D	I	E	B	S	N	T	-There

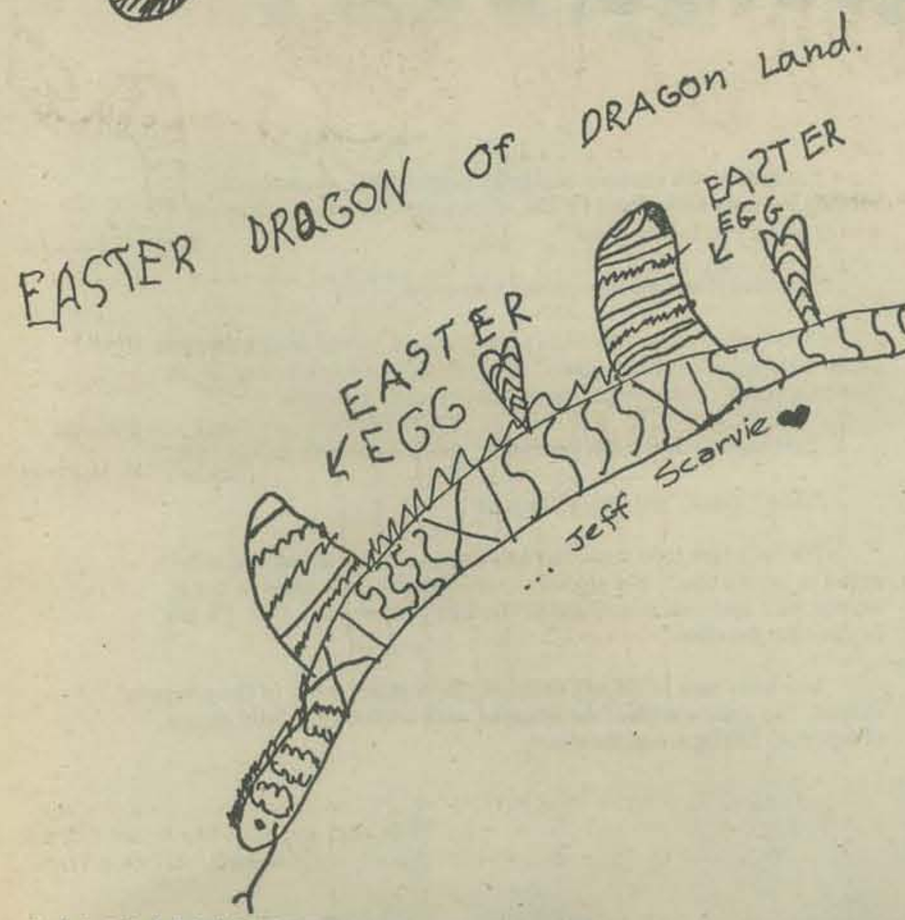
S

START AT H.
1. move to adjacent Boxes (N, S, E, W) to SPELL A WORD.
2. AT THE END OF A WORD, use THE LAST Letter to form a WORD at right angles To the completed word.
3. ALL WORDS ARE READ NORTH TO South OR west to EAST, but may be formed in an OPPOSITE direction.



EAZTER SOLUTIONS: EEE S EE S EE SSSS
: EEE SSESSSSEEE





Pam Jensen
Debbie Robinson
Donita Goodman
Sonja Jellison
Colleen Gilpin

N

Here →

H	1	0	2	1	4	2	1
0	4	2	0	0	2	0	2
1	0	2	1	3	1	2	1
0	1	0	2	0	2	1	2
4	2	1	1	4	1	2	0
0	1	2	2	1	2	0	1
2	1	0	4	1	1	2	0
1	2	1	0	2	0	1	T

W E

← THere

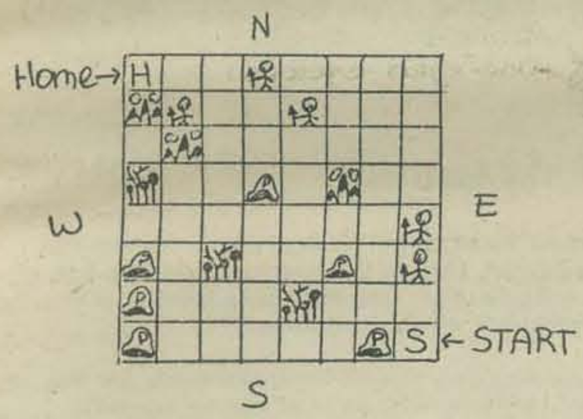
S

Ford Mustang	20 steps=21 POINTS
Mazda	14 steps = 20
Porche	16 steps=14
Plymouth Duster	18.steps=23
Datsun	20 steps=24
Audi Fox	20 steps=29
V.W. Bus	20 steps=23
Toyota	18 steps=22
Volvo	24 steps=29
Hornet	18 steps=21

Rules

1. Determine what kind of car you want.
2. You have to use the number of steps to match the total that's given above for each car. You can only go up and down and sideways. You have to go from Here to THere.

Lance Robinson
Marie Born
Dave Fuhrer
Mike Gangle



1. START AT S.E. CORNER (START).
2. FIND WAY home in 33 moves.
3. ONLY MOVE THROUGH ANY SQUARE 1 TIME.
4. YOU MAY MOVE HORIZONTAL OR VERTICAL.
5. DON'T MOVE THROUGH OCCUPIED SQUARES.
6. COUNT HOME AS 33 SQUARE.

KEY

MTS. -
SAVAGE -
JUNGLE -
PIRATE -

PROGRAM: NWWSWWNWWNNEEEEE-NENNNWWSSWWNNWW.

Sue Lovelady
Claudia Doiel
Karon Ruffin
Kathy Swagerty

Here →

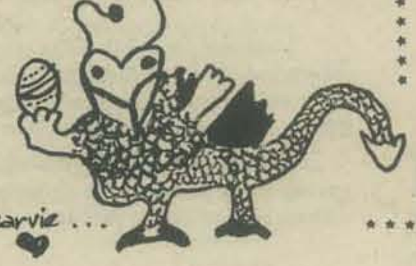
H	m	m	B	C	C	B	m
m	B	C	B	C	m	C	m
m	B	C	m	B	m	m	m
B	B	m	C	C	C	B	C
B	C	B	m	B	C	B	C
m	B	C	C	m	C	m	B
B	C	B	B	B	C	C	m
m	B	m	C	m	B	B	T

← There

C = Cat
m = mice
B = Birds

Rules

The object of This game is to avoid The Cats and to limit the number of mice squares that can be visited. Example: move from Here to there, using exartly 7 m's, no C's, and any number of B's.



last installment

A FANTASY OF

On a Thursday night, sometime in the late 1970's, I visited the People's Computer Center of Menlo Park. They now had two storefronts, and one was open that night for classes. The other, called THE ANNEX, is used for office and meeting space.

Last issue, we sat in at THE ANNEX for the weekly community meeting. For this issue, I visited their funky computer recreation center to talk to the people there and watch the classes.

Here's my story about that Thursday night, sometime in the late 1970's . . .

Time machines being what they are, you can't blame me very much for my annoyance after the last trip.

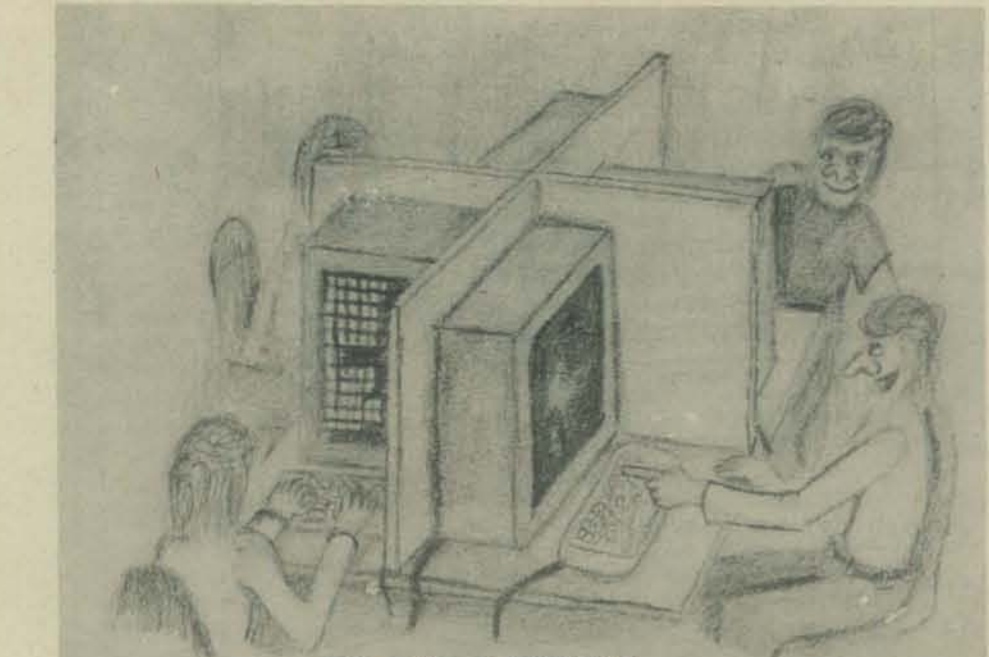
I had set the dials for September 1989, hoping to visit the storefronts here in Menlo Park *after* the opening of the Cybernetic Theatre. The rotary thumbwheel for the tens digit must've been malfunctioning, and I landed with the gentle *Hissssss* of released temporal energy in 1979.

It was nighttime, and since the circuitry of my time machine (cleverly woven into a pair of suspenders) needs about two days of sunlight to recharge, I decided to stay.

Remembering my history (futury ?), I recalled that there were two storefronts by this time. The newest addition, THE ANNEX, was closed. So I wandered into the Center.

It was packed, and things were jumping. There was a high partition in back, behind which I would find the hardware supplies and work area. To my left was Games Alley, to my right a huge pile of oversized pillows.

There were several small crowds of people, each surrounding a large table. I could see 4 or 5 of the tables, and each had 4 large tv screens with keyboards, joysticks, and many had a funny little box.



TV SCREEN TERMINALS

In the back of the room, just on my side of the hardware partition, I could see more screens, one of which was in full color. I started walking that way.

"Hi. Can I help you?"

I turned around to find the voice. She was a little shorter than I, and obviously one of the 5 full-time teachers the Center should have by 1979. Her name was Jane.

"Ummmm, yes" I said, pausing to think fast, "I'm a reporter for a small paper. I'm sure you've never heard of us -- we're located far away." True enough, 1979 *is* far away from 1975!

I'm here by the strangest accident" I continued, unconsciously rubbing my suspenders, "and I'd like to look around, maybe even write a story about you for my paper."

This time it was she who paused a moment.

"Uh, well . . . sure. Why don't you just look around during the class. Daren is the other teacher tonight." She pointed out a tallish, lean fellow with trim beard. "Afterwards, we can answer some questions.

"But before you publish your story, check it out with Harold. OK?"

"Sure," I said, "but which one is he?"

"He isn't here right now. But he's always running in and out,twice as fast as anyone else." She giggled a little. "Maybe *four times* as fast as anyone else. Just wait around and you're sure to meet him. OK? I'll talk to you after the class."

Jane went back to helping someone play a game on one of the computer screens. The game was Hunt the Wumpus, with occasional ghostly images of superbats flitting across the screen.

THE PILLOW FIGHT

The class, I noticed, was made up of people of all ages. And everyone was doing something different. Here and there tv screens showed the dark of interstellar space, with pairs of tiny spaceships spitting bullets at each other. Spacewar, I remembered, was old even back in 1975.

Someone was piloting a space ship, and I stopped to watch. She was about 16 years old and I asked her to explain to me what was happening.

"Oh this is called Space Yachting, my favorite of all the games. You never played it before?

"Well, it's kind of tricky. Tonight, I'm trying to get my space yacht from here," the screen now showed a map of the solar system, "to here and then back. It's like the old game of Lunar Lander 'cause I have to make the round trip to Jupiter with barely enough fuel. I like to use Mars for a whipping point. If I get close enough, its gravity whips me out even faster, so I can save fuel!

"My favorite part is going through the asteroid belt 'cause that's fun."

"Doesn't it take along time to play an entire game?" I asked, trying to sound naive. Popping in and out of the future, I had seen some really far out games that were still ten years in the future!

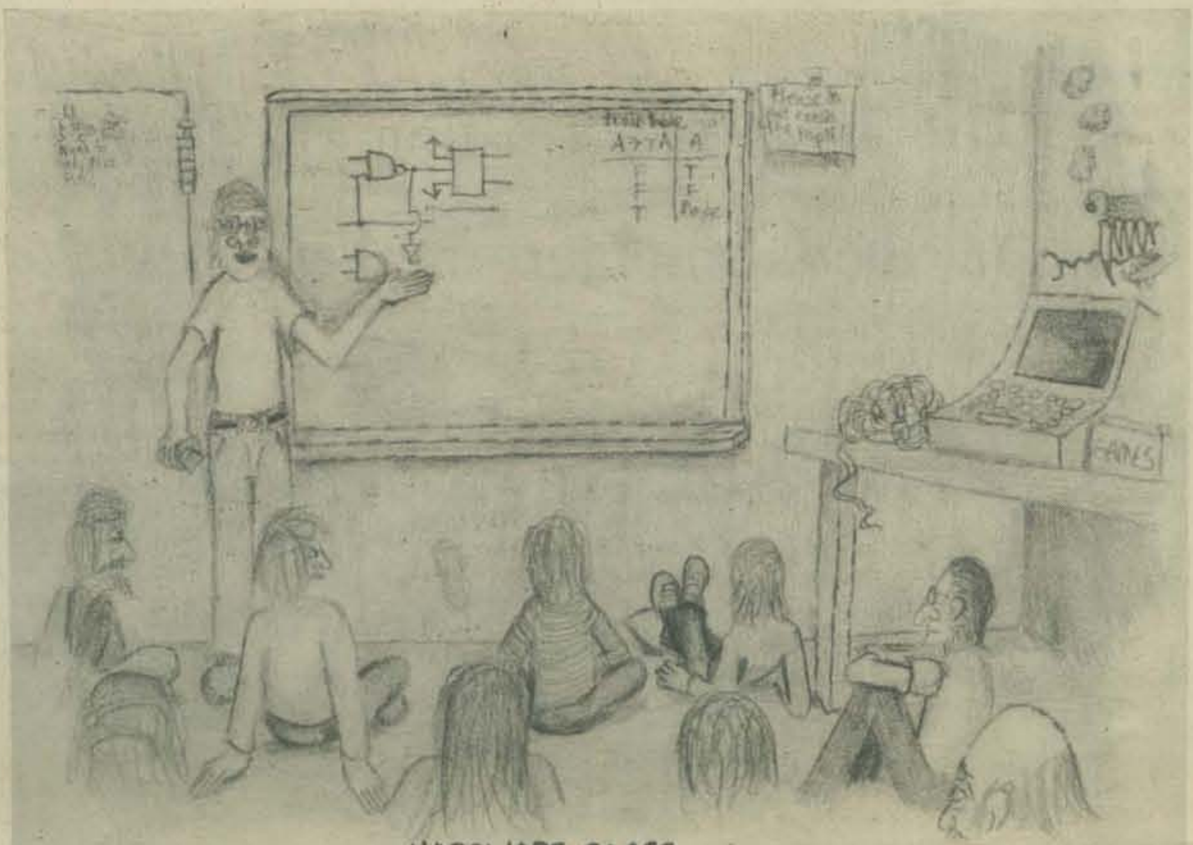
"Oh, that's what I like best. When my yacht gets close to a planet or asteroid, *time slows down* so I can navigate. Then, when I'm going through free space when nothing is happening, *time speeds up* again. The whole game lasts maybe 20 or 40 minutes. That's really a couple of years. Oh, I have to get back. Here come the asteroids!"

I watched for awhile. The screen showed the forward view, and while she worked the joystick and keyboard, two smallish rectangles in the bottom of the screen were flashing radar blips, fuel indicators, meteorite damage reports, crew health and life support system details. She was really into it.

In fact, each person in the whole room was into *something*. There was a group in the pillow corner enthusiastically slugging each other with the huge, clumsy, hard-to-handle pillows.

FUTURE FORMS part 4

GAMES ALLEY



HARDWARE CLASS

By now, my ears were getting used to the constant din. I was beginning to hear some strains of music. In the back, near the partition, I could see a sound system with record player, dual tape decks, etc. etc. Nearby the table was a cabinet with drawers and drawers, most of which had labels : **TAPE LIBRARY, CASSETTES, COLOR GRAPHICS, COMPUTER ART MOVIES, SOUNDTRACKS, GAME SOUND EFFECTS, NATURE NOISES, SPACE SOUNDS.**

When I listened closer, I could hear several distinct musical strains. That's when I found out what the *funny little boxes* did. Each was a tune box, under computer control. It was an independent processor! A game could load up its box with half a dozen tunes, then with a single command, start an entire tune to coincide with a development on the screen. Action on the screen would overlap music from the box.

There was a crowd of youngsters around one of the large tables, listening to the tunes of the four programmers working there. The tallish teacher, Daren, was pointing to some letters on a screen, beating a tempo with his arm.

I walked over and asked where the tune boxes had come from.

"Someone in the intermediate hardware class invented them for a design project. The idea was simple: separate music, visual and other effects into separate processes that can run simultaneously and communicate. Did you want to talk to the inventor? His name is Marty, and he's teaching tonight's hardware class. "

I thanked Daren and wandered back. looking for the door through the partition. Funniest thing, as I walked into the hardware area, my suspenders began picking up some buzz. I flipped off the "absorb power" catch, reminding myself to turn it back on tomorrow.

A young fellow walked up and said, "Hi. I'm called Marty, although my real name's Dave. Can I help you?"

"I hope so. Could you tell me in general what's happening nowadays? I'm visiting from far away, but I know about computer hardware and software."

"Sure. I've got a few minutes before checking out the class. Let's see. We have 3 levels of classes: introductory, like tonight, were you learn how to use basic circuits in design; intermediate, where you build a couple of projects of your own design, and independent study, where you cover some theory and work on one large project. About 30 people are taking intro design classes; the intermediate group is about 20, but we've only got a dozen or so doing independent projects. Did you visit Games Alley? Many of the really good projects end up there, unless they want to take theirs home.

"Designing and building games is pretty popular -- that's the nature of the place, I guess, but everyone wants to build their own computer the first week."

Just then, someone came racing past the open door, towards the bathroom. A few seconds later, he cam back, slowed to a stop, and shouted, "David! I haven't seen you since ... "

"Hi, Harold," I said, grinning a grin. "It's a long, long story"

And I was fingering my suspenders as I told Harold about traveling in Europe for the past 4 years.

FELLOW FANTASISTS

This is the last planned installment of Fantasies of Future Forms. Write to us & tell us what you think the future will be (should be ?) like. In fact ...

*** SHARING ***

In this universe of fantastic sentients, there must be other "Fantasies of Future Forms." Do you have one? Are you in touch with life on a different plane? In a parallel worldstream? Send it, please – if we publish it, we will – in turn – share by giving you a free 1975-76 subscription. Please tell us where in the universe to send it.

Text: David Kaufman
Graphics:Albert Bradley

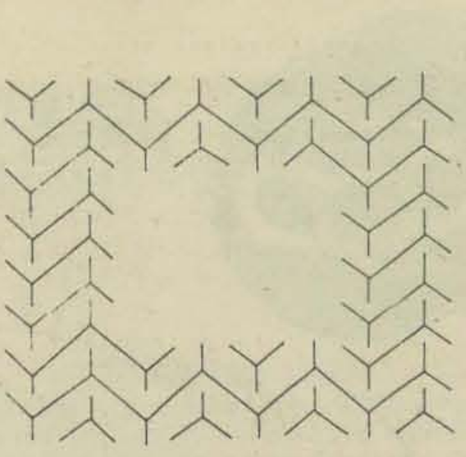


Figure 1A

"...In our work we are examining the potential for producing variation in design through plastic deformation. Plastic deformation is an application of force to produce a change of form in mass. When a plastic deformation is applied to a static design, there is a record of the force (its amount and direction) reflected in the image. The energy necessary to produce the force is imparted to the design, making it kinetic. See figure 1A (a design at rest) and figure 1B (the same design after deformation)..."

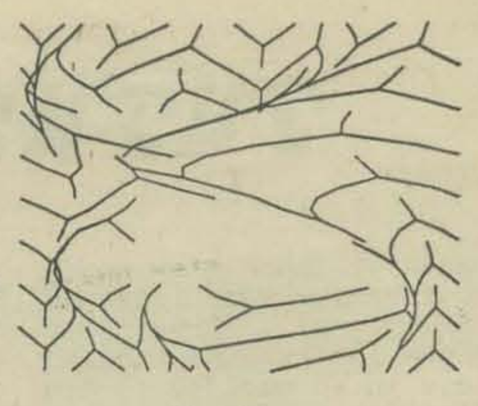


Figure 1B

Excerpts: The *plastic* Deformation of Design: An Applicati n of the Computer in Art

by Jay Leavitt and *Ruth Leavitt*

" .. Optical artists and for that matter most computer artists begin their work on a grid. We do too. Units create designs. Vasarely and Riley manipulate each unit by hand. Most computer artists manipulate the unit by formula. See for example [7]. We are not formula dependent. Nor do we consider each unit individually. Only after a pattern is established does our work begin. We rearrange the design by applying force to the mass. Each sketch the artist makes allows the piece to expand or contract as a whole. The goal is to change the static design into a kinetic one reflecting the plastic event and incorporating the energy necessary to produce it.

Methodology

Imagine a sheet of rubber. The edges of the sheet are clamped down; the center has freedom of movement. A pattern is repeated about the edge of the sheet leaving a blank area in the middle. The rubber sheet is pushed and pulled, thus distorting the original design and creating a totally new composition. By deforming the initial design in different ways a series of variations on the same design results. In addition to its aesthetic qualities, a pattern is selected for its ability to function as a vehicle for movement and color. The deformations are realized by means of an interactive FORTRAN IV computer program. Before a particular series of deformations is begun, the pattern must first be represented in the machine by an ordered sequence of points which are stored in a data statement. The data statement was chosen so that the artist could produce a series of deformations based on the same pattern over a long period of time without having to repeatedly reenter its representation, although the data statement (pattern) could be changed at any time. The pattern itself is a set of straight lines connecting the points. However, it is not necessarily the case that a straight line is drawn between every point and its successor. .."

* Ruth Leavitt ♡
will speak at
DeAnza College, Cupertino
on
April 7 at 2 pm
in Forum 3 ...

Conclusions

" . . . We have been concerned with 'stretching' patterns to achieve variation in design. A model was set up on the computer to allow the artist freedom to examine plastic deformation as a dynamic agent. The artist was not called upon to abandon his traditional role in the production of a work. The gestures of movement input by the artist in the form of vectors govern the calculations made by the machine.

We found that the artist, while making sketches of movement, proceeds in much the same manner as an action painter. Just as Pollock would control the amount and direction of paint on the canvas, so too the artist using this method controls the amount and direction of force into the pattern. Like the abstract expressionist, the computer artist also finds that 'nice accidents' occur. Each work begins its life in complete static order, devoid of emotion. It becomes humanized as the artist imparts energy to the mass. However, the orderliness of the 'concrete school' is not lost in the process of making the piece kinetic. On the contrary, we found that the 'expressionist' use of force intermingles with the 'concrete'patternsforming a harmonious final image.

Thus far we have examined three patterns: the first deals with line (Figures 1,4), the second with mass (Figures 5,6), and the third with a projected three dimensional form (Figures 7,8). It was observed that each design required different methods of stretching in order to obtain a satisfactory composition. Successful stretches applied to one series were rarely successful when applied to any other. Nevertheless, we obtain the same desired effects in all three series. We not only energize the picture, thereby making it kinetic, but we can actually retrace the process of injecting force into each work. In every picture part of the pattern remains undistrubed. This not only allows the viewer to identify with the basic pattern but also provides contrast with the kinetic portions of the picture. This is illustrated by Figure 4. The right hand side of the picture retains the static herringbone pattern. As we progress counter-clockwise we note a gradual elongation of the herringbone pattern ultimately yielding the full gamut of forms it can assume.

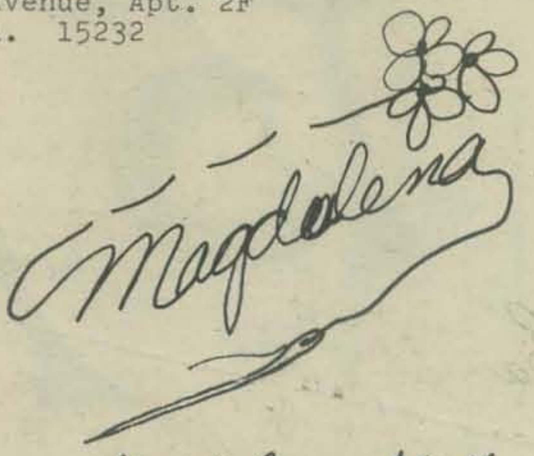
The prismatic series (Figures 5,6) dealing with mass is especially conducive to change. The static pattern naturally yields many forms: triangles, squares, stars, and octagons. When energized the family of forms multiplies. In Figure 5 the triangle, which is a passive component of the static pattern, becomes prominent. Other variants also appear in this picture due to the folding of the rubber sheet upon itself. The assortment of shapes achieved by folding is particularly noticable in Figure 6. The fold forms a Z across the middle of the picture segmenting larger shapes and creating small pentagons, diamonds, rectangles, and elongated triangles. The tension which is evident in the picture is a direct result of the magnitude of opposing stretches which cause the rubber sheet to fold upon itself. Figure 6 also illustrates how the same shape is varied by expanding or contracting its size. The variant shapes assumed by the pattern are reinforced through the use of color. The tones of the colors were carefully selected so that the patterns a viewer would see are not predetermined. The application of color allows the picture to assume its various shapes.

The projected three dimensional series (Figures 7,8) was the most difficult to vary. The massive quality of the pattern would neither allow aesthetically satisfying large deformations nor folds. Great care was taken in preparation of the sketches so that the forms would not collide. In Figure 7 the picture is achieved through a series of small deformations forming a spiral. It is noteworthy that in this series, although there is less distortion and segmentation of the forms, the inserted energy is much more apparent. In both Figures 7 and 8 there is a feeling of compression of the smaller forms by the shifting larger forms. . ."

♡ - thanks to Jillian Quirke for providing us with this paper and information) ... ♡

*** generating patterns from people's names ***

654 Maryland Avenue, Apt. 2F
Pittsburgh, Pa. 15232



Greetings!

We enclose a small program and some sample output for your computer-playing pleasure.

The idea of generating patterns from people's names is known by most handweavers (see, e.g., A Handweaver's Workbook by Heather Thorpe). Drawing patterns by hand to determine if they are interesting enough to actually weave is quite tedious so we decided to test our newly acquired abilities at BASIC (it's more fun than FORTRAN!). The enclosed program was implemented in BASIC-Plus on a PDP-11.

We also have a similar program which generates patterns based on a sequence of random numbers. We will be happy to send you a copy, if you like.

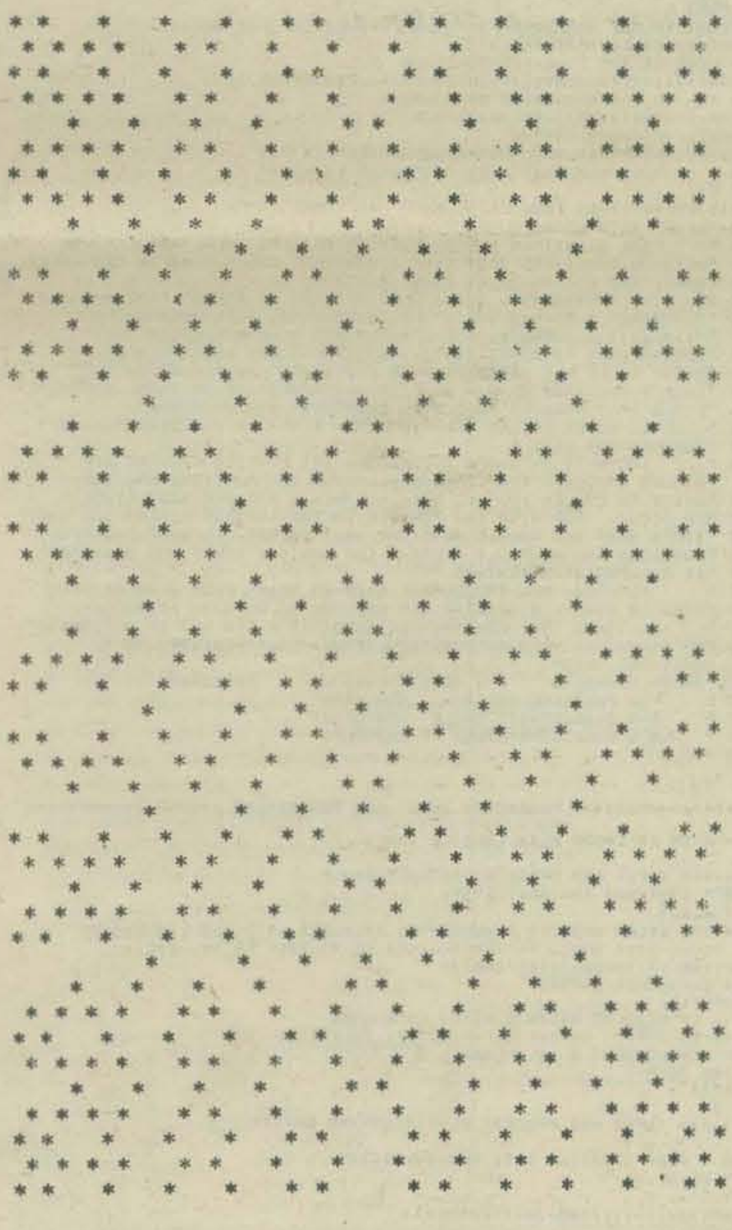
Enjoy!

Richard Mickelsen

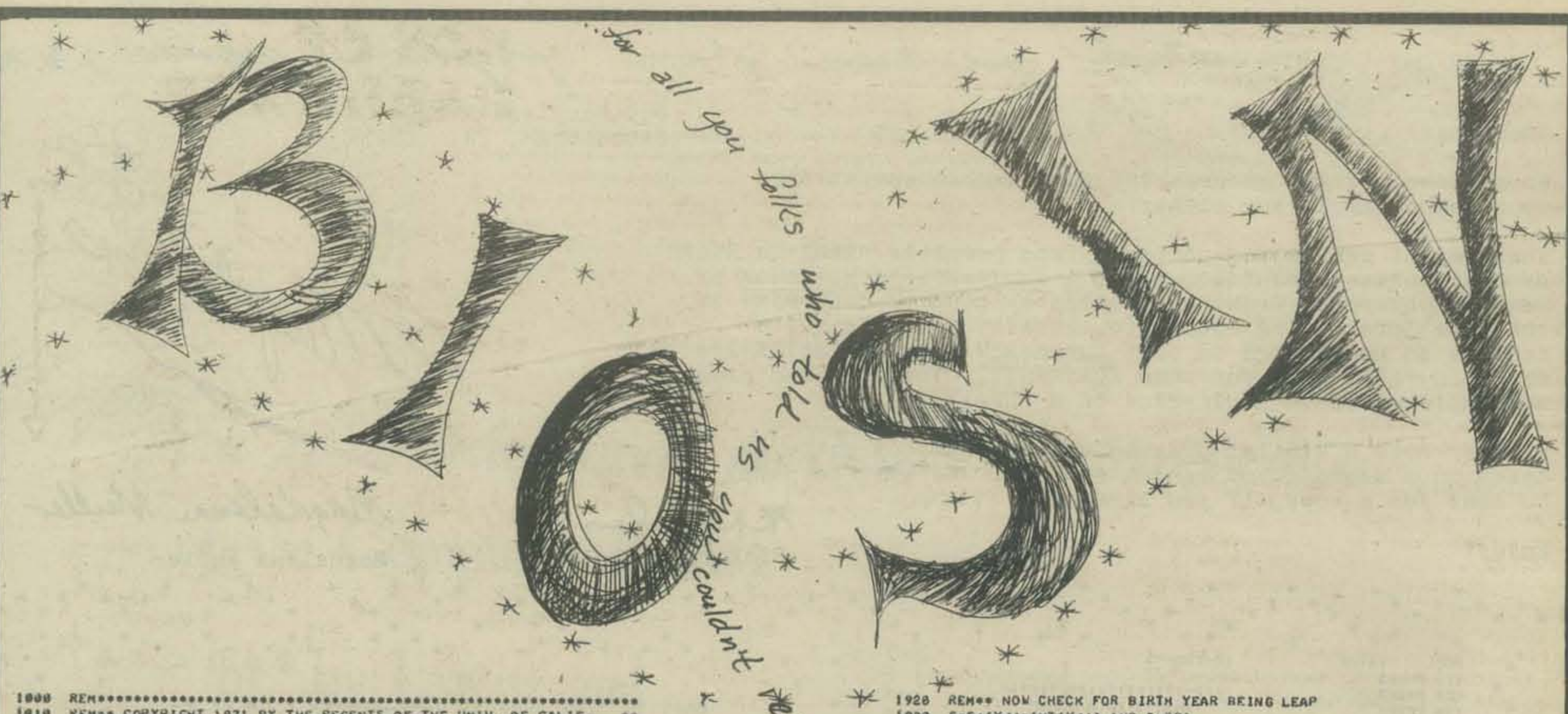
Magdalena Muller
Magdalena Muller

```
PCCI    10:18           13-FEB-75
110 PRINT
120 PRINT "                  ?????????????????????"
130 PRINT "                      WHAT'S IN A NAME"
140 PRINT "                  ?????????????????????"
150 DIM N(20), C(80)
160 PRINT
170 PRINT "THIS PROGRAM WILL GENERATE A PATTERN BASED ON YOUR
180 PRINT "NAME. IF YOUR NAME CONTAINS MORE THAN 20 CHARACTERS"
190 PRINT "INCLUDING BLANKS, ONLY THE FIRST 20 WILL BE USED TO"
200 PRINT "FORM THE PATTERN. SIMPLY TYPE IN YOUR NAME FOLLOWED"
210 PRINT "BY A CARRIAGE RETURN"
220 PRINT
230 PRINT "YOUR NAME, PLEASE"
240 INPUT N$
250 REM CHECK THE LENGTH AND TRUNCATE, IF NECESSARY
260 LET M = LEN(N$)
270 IF M > 20 THEN M = 20
280 LET N$ = LEFT(N$,M)
290 PRINT
300 REM PRINT THE PATTERN NAME
310 PRINT "   <<";N$;">>"
320 PRINT
330 PRINT
340 REM PUT THE CHARACTERS IN A NUMERIC VECTOR
350 CHANGE N$ TO N
360 REM C(0) IS A DUMMY
370 LET C(0) = -2
380 REM J WILL INDEX THE ASSIGNMENTS
390 LET J=0
400 REM CONVERT KEY CHARACTERS
410 LET F1=ASCII("F")
420 LET M1=ASCII("M")
430 LET T1=ASCII("T")
440 FOR I = 1 TO M
450 LET J = J + 1
460 REM MAKE THE ASSIGNMENTS
470 LET C(J) = 1
480 IF (N(I) > F1) THEN C(J) = 2
490 IF (N(I) > M1) THEN C(J) = 3
500 IF (N(I) > T1) THEN C(J) = 4
510 REM BEGIN CHECK FOR ASSIGNMENT VALIDITY
520 REM ELIMINATE REPEATED ADJACENT ASSIGNMENTS
530 IF C(J)<>C(J-1) THEN 590
540 LET C(J+1) = C(J)
550 LET C(J) = 5-C(J+1)
560 LET J = J + 1
570 GO TO 630
580 REM ELIMINATE ADJACENT ASSIGNMENTS DIFFERING BY TWO
590 IF ABS(C(J) - C(J-1))<>2 THEN 630
600 LET C(J+1) = C(J)
610 LET C(J) = 5-C(J-1)
620 LET J = J + 1
630 NEXT I
640 REM REFLECT ABOUT THE END
650 LET N = J
660 FOR I = N TO 1 STEP -1
670 LET J = 2*N-1
680 LET C(J) = C(I)
690 NEXT I
700 LET N = 2*N-1
710 REM CONSIDER THE ASSIGNMENT AS BOTH A ROW AND A COLUMN
720 REM  ASSIGNMENT. ASSIGN A * TO ELEMENTS FOR WHICH THE
730 REM   ROW AND COLUMN ARE THE SAME
740 FOR I = 1 TO N
750 LET M = C(I)
760 FOR J = 1 TO N
770 IF C(J)=M THEN D$="*" ELSE D$=" "
780 PRINT D$;
790 NEXT J
800 PRINT
810 NEXT I
820 PRINT
830 PRINT
840 INPUT "WANT TO TYPE ANOTHER NAME--ENTER Y OR N";A$
850 IF A$ = "Y" THEN 220
860 END
```

```
WANT TO TYPE ANOTHER NAME--ENTER Y OR N? Y

YOUR NAME, PLEASE
? MAGDALENA MULLER

     <<MAGDALENA MULLER>>
```



```
WANT TO TYPE ANOTHER NAME--ENTER Y OR N?
```



for all you folks who told us you couldn't read the BIOSIN listing (Vol.3 No.2 - November '74) → here is a nice clean listing for you

```
1000  REM************************************************************
1010  REM** COPYRIGHT 1971 BY THE REGENTS OF THE UNIV. OF CALIF.    **
1020  REM** PRODUCED AT THE LAWRENCE HALL OF SCIENCE, BERKELEY      **
1030  REM** (MODIFIED AND DOCUMENTED BY THE AARDVARK AT THE PCC)    **
1040  REM************************************************************
1050  REM
1060  DIM A$[21],B$[36],C$[72],A[12],X[3]
1070  REM** A IS AN ARRAY CONTAINING THE NUMBER OF DAYS IN EACH MONTH
1080  REM** A(1) IS HOW MANY DAYS IN JANUARY, A(2) HOW MANY DAYS IN FEB-
1090  REM** RUARY, AND SO ON. FNY IS ALWAYS USED TO DETERMINE HOW MANY DAYS
1100  REM** ARE IN FEBRUARY.
1110  MAT  READ A
1120  REM** FNY IS THE NUMBER OF DAYS IN FEBRUARY OF THE YEAR Y
1130  DEF FNY(Y)=29-SGN(Y/4-INT(Y/4))
1140  REM**  FNS IS THE SINEWAVE FUNCTION
1150  DEF FNS(S)=INT(14*SIN((S/P-INT(S/P))*6.28318)+30.5)
1160  REM** FNM IS THE REMAINDER OF X AND 7
1170  DEF FNM(X)=INT(7*(X/7-INT(X/7))+.5)
1180  A$="MONTUEWEDTHUFRISATSUN"
1190  B$="JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC"
1200  PRINT " DO YOU KNOW WHAT A BIORHYTHM IS [Y-N]";
1210  INPUT C$
1220  IF C$[1,1]="Y" THEN 1550
1230  IF C$="STOP" THEN 2780
1240  PRINT "     THE BIORYTHEM THEORY POSTULATES THAT THERE ARE CERTAIN"
1250  PRINT "METABOLIC RHYTHMS THAT HAVE A CONSTANT CYCLE TIME IN THE HUMAN"
1260  PRINT "BODY, AND ARE KNOWN AS INNER CLOCKS."
1270  PRINT "     TWO PHYSICIANS, ONE A CONTEMPORARY OF FREUD, CONCURRENTLY"
1280  PRINT "DESCRIBED A 23-DAY CYCLE THAT CORRELATED WITH PHYSICAL "
1290  PRINT "VITALITY, ENDURANCE, AND ENERGY; AND A 28-DAY CYCLE THAT"
1300  PRINT "CORRESPONDS TO SENSITIVITY, INTUITION, AND CHEERFULNESS. A"
1310  PRINT "THIRD CYCLE WAS OBSERVED BY A PROFESSER AND OTHERS IN THE"
1320  PRINT "1920'S AND 1930'S, AND IS A 33-DAY COGNITIVE OR INTELLECTUAL"
1330  PRINT "CYCLE THAT RELATES TO MENTAL ALERTNESS AND JUDGEMENT."
1340  PRINT "     ALL THREE CYCLES START UP FROM BIRTH OR THE BEGINNING OF"
1350  PRINT "INDEPENDENT LIFE."
1360  PRINT "     THE DAYS ON WHICH THE MEDIAN (0) LINE IS CROSSED ARE "
1370  PRINT "CALLED CRITICAL DAYS, ESPECIALLY FOR THE PHYSICAL AND SEN-"
1380  PRINT "SITIVITY CYCLES AND ARE WHEN ACCIDENTS ARE THE MOST LIKELY"
1390  PRINT "TO OCCUR.  THE HIGH (+) PERIODS IN THE VARIOUS CYCLES ARE THE"
1400  PRINT "TIMES WHEN YOU SHOULD HAVE THE MOST ENERGY, BE MOST CHEERFUL,"
1410  PRINT "OUTGOING AND MENTALLY ALERT.  THE LOW (-) TIMES ARE REGARDED"
1420  PRINT "AS RECUPERITIVE PERIODS."
1430  PRINT "     ALTHOUGH ALL CYCLES ARE SAID TO START FROM ZERO AT THE"
1440  PRINT "TIME OF BIRTH, BIORHYTHM HAS NOTHING TO DO WITH ASTROLOGY."
1450  PRINT "     A TOKYO TAXI COMPANY HAS REPORTED A 60% CUT IN ACCIDENTS"
1460  PRINT "BY GIVING DRIVERS AN 'EXTRA-CAREFUL-TODAY'REMINDER ON "
1470  PRINT "CRITICAL DAYS"
1480  PRINT "GRAPH SYMBOLS:     I : MEDIAN LINE          (0 LINE)"
1490  PRINT "     P : PHYSICAL CYCLE      (23-DAY)"
1500  PRINT "     S : SENSITIVITY CYCLE  (28-DAY)"
1510  PRINT "     C : COGNITIVE CYCLE     (33-DAY)"
1520  PRINT ""
1530  REM
1540  REM
1550  REM****************** PARAMETER INPUT AND PROCESSING ******************

1560  REM** K IS AN ERROR FLAG [SEE L# 1580]
1570  K=1
1580  REM****** INPUT AND PROCESS BIRTHDATE***
1590  PRINT " BIRTHDAY [MM,DD,YYYY]";
1600  INPUT M,D,Y
1610  REM** THE BELOW REDUCES A NUMBER TO ONLY ITS 10'S AND 1'S DIGITS
1620  REM  [THUS 1973 WOULD BE TRANSFORMED TO 73,1959 TO 59, ETC.]
1630  Y=INT(100.1*(Y/100-INT(Y/100)))
1640  REM** CHECK FOR ERRORS
1650  A[2]=FNY(Y)
1660  IF M<1 OR M>12 OR ABS(M)#INT(M) THEN 2710
1670  IF D<1 OR D>A[M] OR INT(D) <> ABS(D) THEN 2730
1680  REM** NOW CONVERT D TO JULIAN
1690  FOR I=1 TO M-1
1700  D=D+A[I]
1710  NEXT I
1720  REM****** INPUT AND PROCESS PLOT-STARTING DATE**
1730  K=2
1740  PRINT " PLOT STARTING DATE [MM,DD,YYYY]";
1750  INPUT M1,D1,Y1
1760  D0=D1
1770  Y1=INT(100.1*(Y1/100-INT(Y1/100)))
1780  A[2]=FNY(Y1)
1790  IF M1<1 OR M1>12 OR ABS(M1)#INT(M1) THEN 2710
1800  IF D1<1 OR D1>A[M] OR INT(D1)#ABS(D1) THEN 2730
1810  FOR I=1 TO M1-1
1820  D1=D1+A[I]
1830  NEXT I
1840  REM
1850  REM
1860  REM** 'S' WILL END UP BEING HOW MANY DAYS FROM BIRTH TO PLOT DATE
1870  S=0
1880  REM** FIRST, ADD IN LEAP YEARS FROM BIRTH TO PLOT DATE
1890  FOR I=Y+1 TO Y1-1
1900  S=S+(I/4=INT(I/4))
1910  NEXT I
1920  REM** NOW CHECK FOR BIRTH YEAR BEING LEAP
1930  S=S+(Y/4=INT(Y/4) AND D>58)
1940  REM** WE ADD HOW MANY YEARS OLD YOU ARE * 365
1950  S=S+(Y1-Y-1+(D1 >= D))*365
1960  REM** AND FINALLY, WE ADD HOW MANY DAYS SINCE THE LAST BIRTHDAY
1970  S=S+(D1 >= D)*(D1-D)
1980  S=S+(D1<D)*(365-D+D1)
1990  IF S<0 THEN 2750
2000  REM** D3= HOW MANY DAYS FROM THE YEAR 1900 TO PLOT DATE
2010  REM** [IT IS USED LATER TO DETERMINE WHAT DAY OF THE WEEK IT IS]
2020  D3=Y1*365+D1
2030  FOR I=1 TO Y1-1
2040  D3=D3+(I/4=INT(I/4))
2050  NEXT I
2060  PRINT " HOW MANY DAYS DO YOU WANT PLOTTED";
2070  INPUT D9
2080  D9=D9*2
2090  D1=D0
2100  REM
2110  REM
2120  REM**              START OF PRINTING ********************************
2130  PRINT "";TAB(20);"(-)       (0)       (+)"
2140  FOR Q=1 TO D9
2150  REM** C$ IS THE OUTPUT STRING
2160  C$=""
2170  IF D1 <> INT(D1) THEN 2270
2180  REM************** SET UP DATE*******
2190  E=FNM(D3-1)+1
2200  REM** YOU WILL NOTE THAT 1/1/1900 WAS A MONDAY
2210  PRINT A$[3*E-2,3*E];
2220  PRINT D1;
2230  IF D1 <> 1 THEN 2270
2240  PRINT B$[3*M1-2,3*M1];
2250  PRINT Y1;
2260  REM************ SET UP PRINTING****
2270  FOR I=LEN(C$)+1 TO 72
2280  C$[I,1]=" "
2290  NEXT I
2300  FOR I=1 TO 3
2310  REM** WHEN X=1,P=23;X=2,P=28;X=3,P=33; FOR THE SIGNIFICANCE OF
2320  REM** THESE NUMBERS, SEE THE HP GAMES BOOK TEXT ON BIOSIN.
2330  P=18+(5*I)
2340  X[I]=FNS(S)
2350  NEXT I
2360  C$[30,30]="I"
2370  C$[X[1],X[1]]="P"
2380  IF X[1]#X[2] OR X[1]#X[3] OR X[2]#X[3] THEN 2410
2390  C$[X[1],X[1]]="*"
2400  GOTO 2510
2410  IF X[1]#X[2] THEN 2440
2420  C$[X[3],X[3]]="C"
2430  GOTO 2450
2440  IF X[2]#X[3] THEN 2470
2450  C$[X[2],X[2]]="*"
2460  GOTO 2510
2470  C$[X[2],X[2]]="S"
2480  IF X[1]=X[3] THEN 2390
2490  C$[X[3],X[3]]="C"
2500  REM** OUTPUT C$
2510  FOR X=15 TO 72
2520  IF C$[X,X]=" " THEN 2540
2530  PRINT TAB(X-1);C$[X,X];
2540  NEXT X
2550  PRINT
2560  REM**  RESET DATE
2570  D3=D3+.5
2580  S=S+.5
2590  D1=D1+.5
2600  IF D1<A[M1]+1 THEN 2670
2610  D1=1
2620  M1=M1+1
2630  IF M1<13 THEN 2670
2640  M1=1
2650  Y1=Y1+1
2660  A[2]=FNY(Y)
2670  NEXT Q
2680  PRINT ""
2690  GOTO 1200
2700  REM**        ERRORS  ***
2710  PRINT "IMPROPER MONTH"
2720  GOTO 2740
2730  PRINT "IMPROPER DAY"
2740  GOTO K OF 1590,1740
2750  PRINT "PLOT STARTING DATE EARLIER THEN BIRTHDAY"
2760  GOTO 1740
2770  DATA 31,28,31,30,31,30,31,31,30,31,30,31
2780  END
```

SON OF HARDWARE

Don Lancaster
SYNERGETICS
Box 1112
Parker, AZ 85344

Dear Lee:

Found out more about the cassette interface. First, a phaselock loop on output is not a luxury, but its essential. Also cheap and doubles as an input signal source if you use the CMOS 4046.

Secondly, modulating the tape is very critical as you have to switch at the zero crossings to keep the phase from changing between the ones and zeros. Taking the PLL's square wave and RC filtering it and then amplitude switching it does NOT work as the filters delay lets you switch in the middle of a cycle.

Instead am trying routing both ends of the capacitor on the PLL into a CA3080 (after attenuation) to recover a triangle wave and modulate it directly and reference the output to ground. This seems simple and cheap enough. The PLL apparently goes non-symmetric unless you use higher values of timing resistances (above 22K).

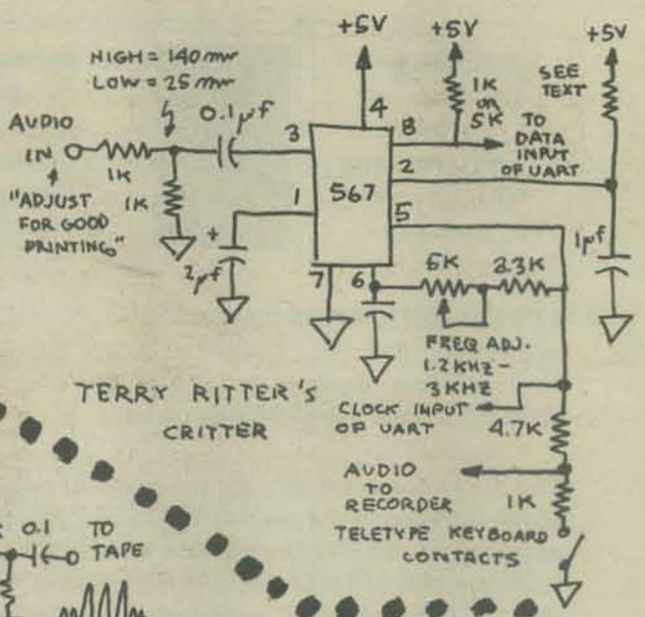
Don't have a final circuit yet, but the beast would probably end up something like this. 4046 as signal source on transmit, PLL on receive. 3080 as modulator. UART as UART. On receiving end, 3080 as AGC. 5558 as Q=3 garbage filter. Second half of 5558 as detector. 4019's as input switching for UART to decide whether or not it comes from keyboard or TVT retransmission. Total IC costs is well under $10 plus the UART.

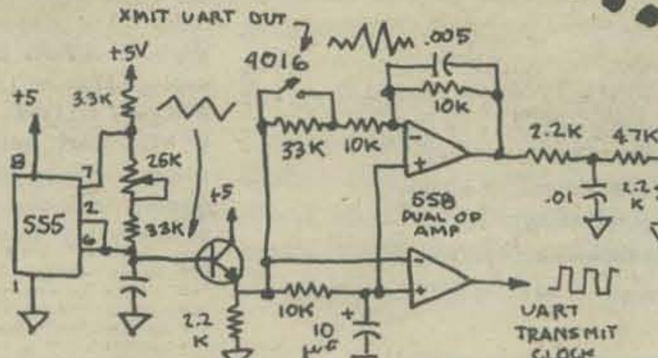
Still do not have the thing working, so we're still with a half breadboarded paper design.

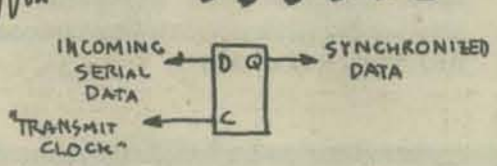
Terry F. Ritter of Dantco (2524B Glen Springs Way Austin, Texas, 78741), submits a design using a 567 with the input signal reduced to a level so that the loop in the chip will stay in lock but the amplitude-sensing output will drop out when the signal goes low. The threshold for this kind of operation is about 50 millivolts, according to the manual. Terry suggests a desensitizing resistor connected to pin 2 of the 567 to allow loop lock with low signal amplitude.

The data sheets, however, seem to indicate that both lowering the signal amplitude and installing the resistor will tend to reduce the width of the loop detection band, the range of frequencies through which the loop will stay in lock. Since the purpose of this design is to track varying frequencies, this seen - as if it could cause problems.

(IS THAT "TEX" RITTER ?)







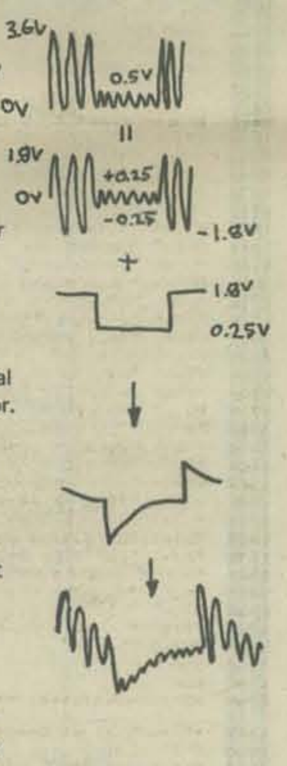
Don Lancaster sent a sketch of work in progress showing how he hopes to lick the zero-crossing switching problem. He uses a 4016 CMOS bilateral switch under control of a "transmit UART". Don is designing his unit to take parallel input, and uses the transmitter section of the UART to generate the serial signal which goes to the tape. This signal will change only when the transmit clock switches, so Don extracts this clock from the carrier oscillator (1760 Hz for 110 Baud). If he used the square wave output of the 555 it would switch at the "corners" of the triangular wave. So he picks off the triangle with an emitter follower and generates the DC average of that wave with the 10k – 10uf filter and uses that as a reference voltage. When the triangle crosses that reference voltage it switches the clock signal to the UART. If you are using a serial input, I would suggest a 4013 D flipflop to synchronize the signal (see sketch). A 7474 would be used if you are using TTL stuff.

NOTHING ON THIS PAGE WORKS!*

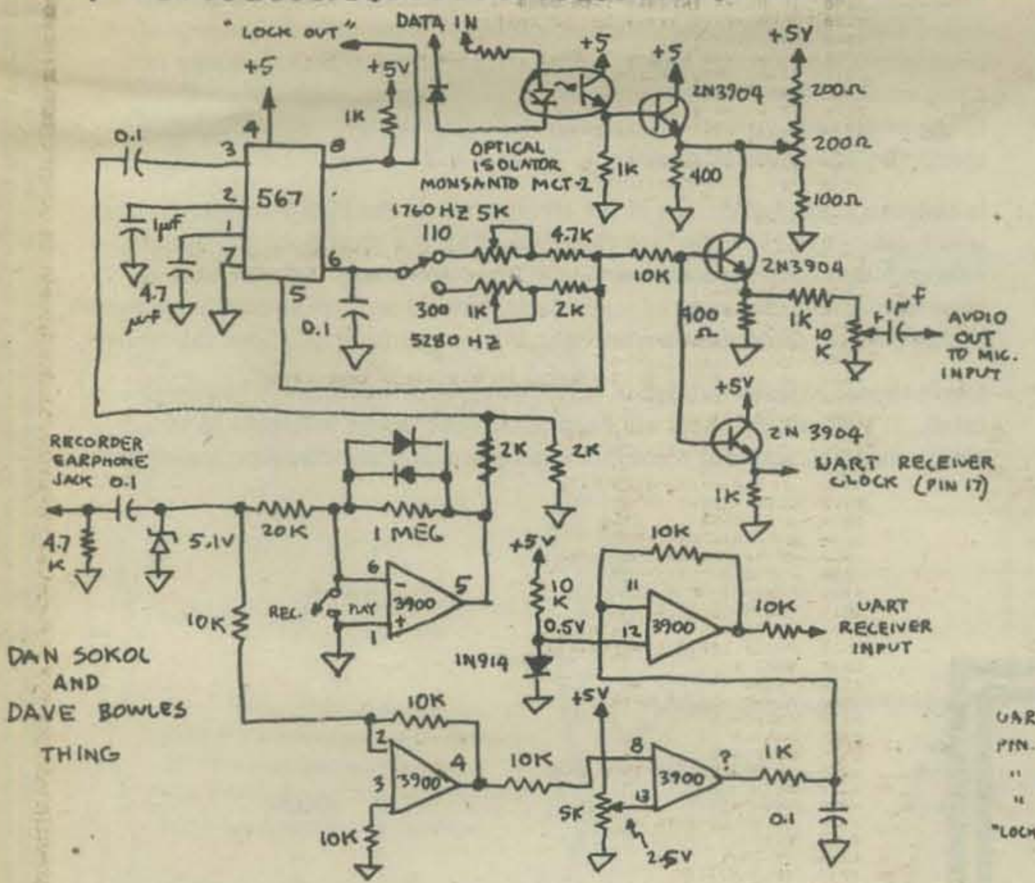
Dan Sokol and Dave Bowles (211 Fall Creek Drive, Felton, Ca. 95018) likewise suggest a 567 as a loop, but do not rely on it for the amplitude sensing. They use a 3900 quad operational amplifier to extract the clock by limiting a highly amplified signal and then shipping it to the loop. The data is sent to a buffer (1 to 1 amplification) and then to a comparator, where an input greater than 2.5 volts causes an output. The half-cycles that come out of the comparator are then filtered by an R-C network and sent to another comparator, which triggers when the slowly-varying voltage crosses it own threshold of 0.5 volts (the drop across the 1N914 diode).

The encoder is an emitter follower with the data signal connected to raise or lower the voltage on its collector. The minimum level is adjustable by the 200 ohm potentiometer. As a result the "bottom" of the output signal is always at ground, and the "top" varies from 3.6 to 0.5 volts. From bitter personal experience with an audio burst encoder I can testify that this will cause trouble. The trouble is that the output signal is really two signals combined — a sine ware with 3.6 and 0.5 volt peak-to-peak amplitude, and a DC square wave which shifts with the data from 1.8 volts to 0.25 volts.

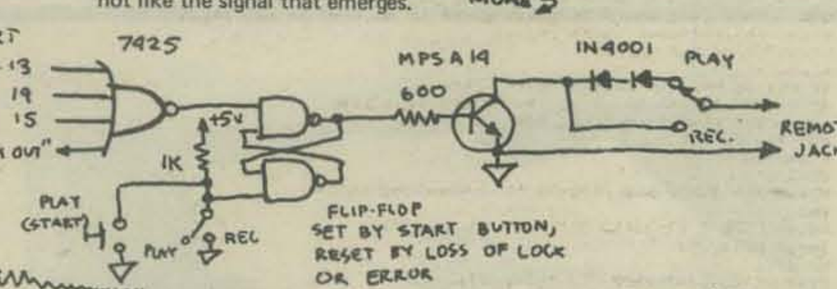
The recorder will, I predict, treat the sine wave signal very well. But the square wave will be warped by the low-frequency response characteristics. (See the illustration for further details). The comparators may not like the signal that emerges. MORE



*-YET!



Dan and Dave do make a suggestion that is most pertinent, though. They understand that the solution to avoiding data overrunning the serial re-transmitter (little problem No. 1 from last time) is not to change the number of stop bits, but to slow down the tape recorder on playback. They suggest inserting two diodes in series with the Remote Control line on Play, none on Record. Alternately, they suggest placing a cocktail straw over the recorder capstan on playback to increase its diameter and slow the tape.



I want to suggest the use of an LM370 as an AGC amplifier and modulator. Its gain changes from 40 db to –10 db as its control voltage goes from 2.0 to 2.6 volts without shifting the DC level of the signal. You tie both inputs up to the positive supply with equal 10k resistors and feed it an input no greater than 100 millivolts peak-to-peak. It has two control inputs (the more positive one controls) and a pin where you can connect a capacitor to slow down the gain-control action. Supply voltage can be from 4.5 to 24 volts, positive only. It's available from most surplus dealers. Data sheet is page 5-1 of the National Linear Handbook.

THIS WAS THE SECOND in a never-ending series which is an attempt at developing a process of Participatory Design of computer hardware. See Vol. 3 no. 3 for the beginning of this design. There is no space left to explain further. SEND design suggestions and refinements to me at →

Lee Felsenstein
PCC, Box 310
Menlo Park CA 94025

book reviews

Both Sides of the Necessary Paradox (Conversations with Gregory Bateson)

Fanatic Life and Symbolic Death Among the Computer Bums

II CYBERNETIC FRONTIERS
by Stewart Brand
Random House - Bookworks 1974 pp. 96
$2.00 (Available from PCC Bookstore)

This little book is actually the expansion of two articles by Stewart Brand which appeared in the *Harpers* 1973 and *Rolling Stone*, 1972. The first is an interview with Gregory Bateson whose ideas the author feels " will indirectly inform damn near everybody's lives." The second, an article about "Computer Bums" and spacewar (no listings, however). It's an entertaining history of Bay Area computer people including Stanford AI, ARPA and Resource One.

- To *want* control is the pathology, not that the person gets control, because of course you never do.
- *Job's sin is that he's pious. . . . The correction for piety is natural history.*
- *You cannot induce a Pavlovian nervous breakdown in an animal out in the field.*
- *All* differences are things of the mind.

excerpts

HOW MANY SPACE MINES DO YOU WANT?

- One of the hackers wrote a program called "The Unknown Glitch", which at random intervals would wake up, print out I AM THE UNKNOWN GLITCH. CATCH ME IF YOU CAN, and then it would relocate itself somewhere else in core memory, set a clock interrupt, and go back to sleep. There was no way to find it.
- *Computer function is mostly one-track-mind, in which inconsistency is intolerable. The human mind fun:tions on multiple tracks (not all of them accessible); it can tolerate and even thrive on inconsistency.*
- Ready or not, computers are coming to the people.

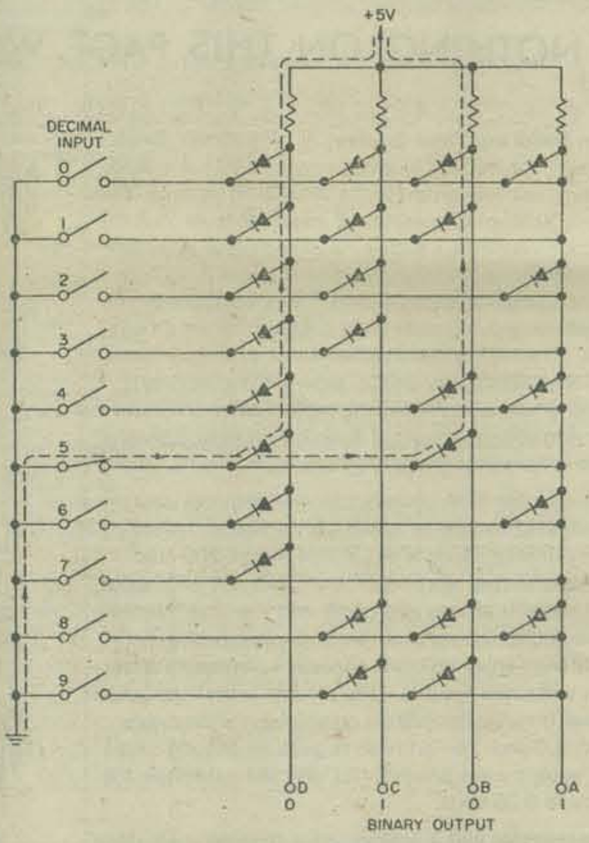


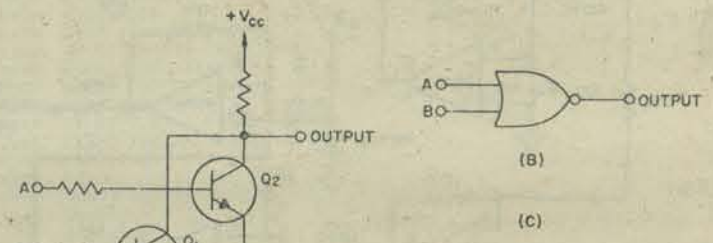
Fig. 3-11. A decimal-to-binary encoder matrix using diode logic.

FUNDAMENTALS AND APPLICATIONS OF DIGITAL LOGIC CIRCUITS
by Sol Libes
Hayden Book Company, Inc., 1975 pp. 192
$5.98 (Available from PCC Bookstore)

This book was intended to allow electronic technicians to learn the basics of digital logic. It starts by assuming that the reader has some knowledge of electricity and electronic basics, and goes from there to provide a very rich, compact collection of information on semiconductors, logic and logic classes, binary arithmetic, memory types, data communications, measuring instruments and analog-digital devices.

It contains some description of the architecture of the PDP-8 computer, but is not meant to go very far into that level. This is a great book for use as a reference by people who are learning digital electronics. It doesn't tell all there is to know about any of its subjects, but it does have enough information in one place to de-confuse anyone who is trying to find out about this subject.

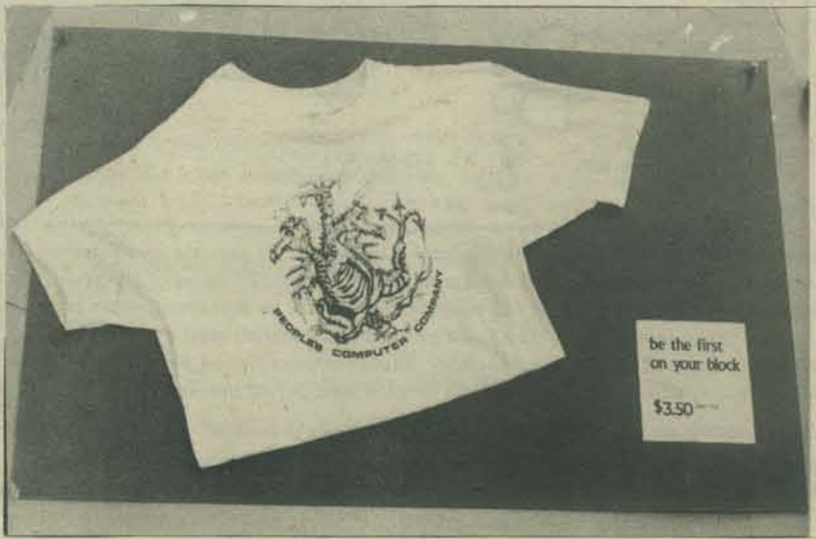
Don't expect to skim through it, or to read it in one sitting and get up educated. It's a book that you will keep going back to and will want to take along when you seek out some more complete information or explanations.



A	B	Output
0	0	1
1	0	0
0	1	0
1	1	0

Fig. 3-2. A 2-input NOR gate: (A) circuitry, (B) logic symbol, and (C) truth table.

Lee Felsenstein



be the first
on your block

$3.50

PCC Bookstore

BASIC, Albrecht, Finkel & Brown, 1973, p. 323 – $3.95
BASIC BASIC, James Coan, 1970, p. 256 – $5.95
BASIC PROGRAMMING, Kemeny & Kurtz, 1967, p. 145 – $6.95
COMPUTERS & COMPUTATION, Scientific American, p. 280 – $6.00
COMPUTER LIB & DREAM MACHINES, Theodore H. Nelson, 1974, p. 186 – $7.00
DRAGON SHIRTS, Nancy Hertert, 1974 – $3.50
GAMES, TRICKS AND PUZZLES, Wallace Judd, 1974, p. 100 – $2.95
GIMME SOMETHING TO FEEL, Jane Wood, 1973, p. 125 – $$2.95
MATH WRITING & GAMES, Herbert Kohl, 1974, p. 252 – $2.45
MY COMPUTER LIKES ME, Dymax, 1972, p. 64 – $1.49
101 BASIC GAMES, Ed. David Ahl, 1974, p. 250, – $5.00
PROBLEMS FOR COMPUTER SOLUTION, Gruenberger & Jaffray, 1965, p. 398 – $7.25
PROFESSOR GOOGOL, Sam Valenza, Jr., 1973, p. 144 – $3.25
SERIOUS GAMES, Clark C. Abt, 1970, p.176 – $5.95
SELLING WHAT YOU MAKE, Jane Wood, 1973, p. 111 – $2.25
STEPS TO AN ECOLOGY OF MIND, Gregory Bateson, 1972, p.517, – $1.95
TTL COOKBOOK, Don Lancaster, 1974, p. 328 – $7.95
II CYBERNETIC FRONTIERS, Stewart Brand, 1974, p. 96 – $2.00
WHOLE EARTH EPILOG, Stewart Brand, Editor, 1974, p. 318 – $4.00
➡ WHAT TO DO AFTER YOU HIT RETURN, PCC, 1975, p. 158 – $6.95
A PRACTICAL, LOW-COST HOME/SCHOOL MICROPROCESSOR SYSTEM
reprint from COMPUTER magazine, 1974 – $1.00

RENEW NOW

your subscription ends with the next issue

order form

BILL TO:

____
Name
____
Address
____
Zip

SHIP TO:

____
Name
____
Address
____
Zip

Purchase Order No

Date

Quantity	Description	Price Each	Total Price

Order Total	
Sales Tax	
Shipping Costs†	
Total Amount	

† $0.50 for order under $10.00
$1.00 for orders $10.00 and over

Mail order to:

People's Computer Company
P.O. Box 310
Menlo Park, Ca. 94025

Give someone you love a
GIFT CERTIFICATE
to the People's Computer Center

$1 – 1 Friday Games Night
$1.50 – 1 Saturday hour on the computer
$5 – 1 month class for a 5-8 year old
$12 – 1 month class of Beat the Computer
$20 – 1 month class in programming

or any combination of the above

ORDER FORM

SEND CHECK OR MONEY ORDER TO:

People's Computer Company
PO BOX 310 • MENLO PARK • CA 94025

MY NAME ____________________
MY ADDRESS ____________________

□ regular subscription

□ student subscription

□ group subscription → □
HOW MANY?

□ back issues : any mix (name your mix)

*□ Renew my subscription for 75 - 76 for only $3.

*□ Send PCC (74 - 75) to a friend for only $3. Please include your name above and friends name below OFFER EXPIRES APRIL 30 '75

FRIEND'S NAME ____________________
ADDRESS ____________________
____________________ ZIP

1 People's Computer Company is a not -for - profit corporation. The newspaper is about recreational and educational uses of computers. . . computers for everyone.

PCC is published 5 times (and sometimes more) during the school year. Subscriptions begin with the first issue in the fall.

2 If you are an elementary or secondary school student you can subscribe to PCC for $3.00. Send cash, check, or money order. No purchase orders. Use your HOME address! Please send us some evidence that you are a student.

3 Single subscriptions are $5 for 5 issues. ($6 outside U.S.A. - surface mail; $12 - air maii)

4 Group subscriptions (all mailed to the same address.)

2 - 9	$4.00 each
10 - 99	$3.50 each
100 or more	$3.00 each

5 Get back issues while they last at the following low low prices:

Vol . I Nos. 1 - 4*	$2.00
Vol. II Nos. 1,3,4,5,6*	$4.00

* Sorry, we are out of Vol. I No. 5 and Vol 2 No.2.

Or mix up individual issues:

2 - 9	$.80 each
10 - 99	$.70 each
100 +	$.60 each

6 Buy a dragon shirt . . . See PCC Bookstore, pages 26 and 27. (support your PCC Dragon)

Your Subscription Expires with the next issue !

RENEW NOW

ONE MORE TIME → 7 Renew my subscription for 1975 - 76 for only $3.00 ****!!!

Send P.C.C. Volume 3, 1974 - 75 to a friend for only $3.00 *

WHO DID THIS ISSUE?

EDITOR: Bob Albrecht

PRODUCTION EDITOR AND MANAGER: Mary Jo Albrecht

CIRCULATION: Lois Britton

ART EDITOR: Pam Scarvie

CONTRIBUTORS:

Bob Albrecht
Lee Felsenstein
Hal Singer
Fred Moore
Dennis Allison
Mary Jo Albrecht
David Kaufmann
Ruth Leavitt
Lillian Quirke
Pam Scarvie
Richard Mickelsen
Magdalena Muller
Paul Hoggatt and friends

TABLE OF CONTENTS