PEOPLE'S COMPUTER COMPANY

Vol.4, No.1 July 1975

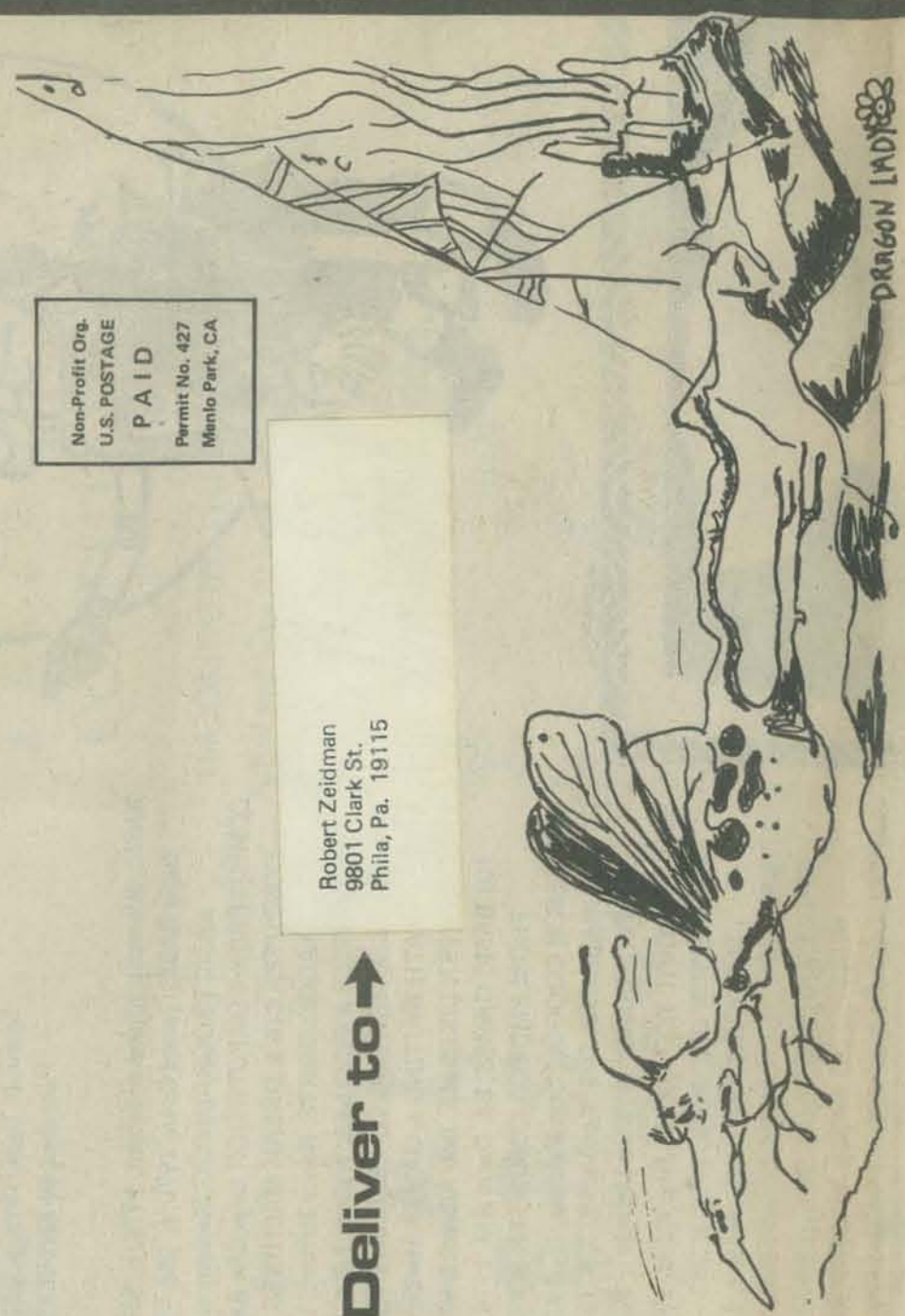
J. Reeve Evan (Jacob Felsenstein)

Hardware Issue

PCC
p.o. box 310
menlo park, ca. 94025

Deliver to →

Robert Zeidman
9801 Clark St.
Phila, Pa. 19115



PCC News & New Stuff

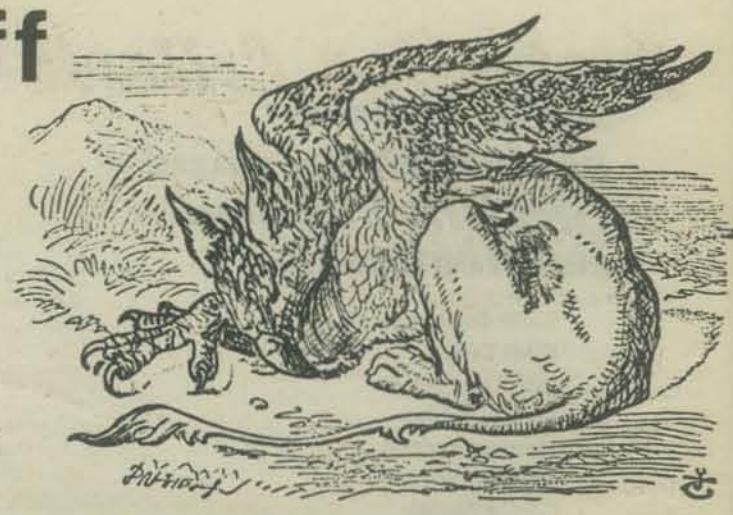
THE NEW DRAGON is nameless no longer, and is growing fast. It's called Community Computer Center, is incorporated, granted tax exempt status in California (Federal application pending) and has two new rooms. One is for hardware classes, the other for a whole lot of DEC equipment. Currently, OS8 is being run with DecTape but soon, when the PDP 11 comes.....
You can still buy the PCC games book from either the PCC bookstore or from CCC, but if you decide to buy from CCC remember to make your check to Community Computer Center - saves trouble at the bank.

MEANWHILE, BACK AT PCC.....

The year of the home computer is here. Clubs and centers mushroom across the nation (8+ with 700+ members in California alone) and the demand for information grows accordingly. So too does the volume of information available. Suddenly, thousands of inventive minds have hardware to play with and the output of ideas is staggering. PCC exists as a medium through which these ideas can be transmitted, some of them..... So we are setting up a system. If you produce a program/design/article/booklet and you want to publish it through us, send it. We will catalog it and make Xerox copies available through the bookstore at about a nickel a page, plus 50cents handling and postage. If an author wants a royalty, the cost will be more - but we are not very interested in handling much that is not in the public domain. Our present staff is four part-timers and the hassles of record keeping for lots of small royalties would be too much for us. But if we feel something really should be made available and paying a royalty is the only way, then so be it.
We have also discovered that there are many valuable items, for example a DOS BASIC for the 8080, which aren't being published because the designer is afraid of his phone. If enough people call you to ask questions, you can't get any work done. MITS receives around 1000 phone calls a day - that's 2 per minute - and has to employ people just to answer phones. If you have something you would like to contribute to the world but don't want phone calls - send it to us. We will publish it and answer the phone. If we can't answer the question, we will try to find someone who can, and if we can't – sorry.

LSI 11 OEM BUY

We wondered (in print) whether there would be enough interest to support an OEM purchase of 50 LSI 11s. This generated about the same number of enquiries but things have happened. We ran into a legal snag, many enquiries were for business use and we are Non-profit. We also heard that Bill Godbout was planning an MSI 11 kit for release in October. It is hoped that it will run the full PDP 11/40 instruction set and be conservatively clocked at 300ns. So we are putting the people who enquired, in touch with another possible OEM buyer and waiting to see just how good the MSI 11 turns out to be.... and it looks good so far.



OUT THERE IN THE REAL WORLD

The first 16 bit computer kit is being produced, and you can have one free! To launch it, Bill Godbout is running a competition. To enter, send your idea of a good name for the beast, the name of the 'secret computer co.' which makes the chip and 25 words or less saying why you should have one. 1st prize is THE CHIP, 2nd an 8080, 3rd an 8008. And if your name is the one used for the kit - you get the whole kit. All entries must be postmarked by 1st Aug 75 and received by Bill by the 10th. They must be marked 'PCC Competition' and all entries become the property of Bill Godbout Electronics.

1ST PRIZE: 16 BIT MICROCOMPUTER CHIP!

CONTEST!

GODBOUT

BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

2nd prize: 8080 cpu

3rd prize: 8008 cpu

We were 1st to offer the 8008 to hobbyists over 16 months ago; now we're setting the pace again with a powerful new 16 bit microcomputer IC in a 40 pin DIP, made by: the SECRET MICROCOMPUTER Co!

YOU MAY WIN ONE OF THESE CHIPS --- SIMPLY:

1) Reveal the Secret Microcomputer Co.'s true identity
2) Tell us in 25 words or less why you should receive a free chip

My Name: **Address:**

Zip: **Phone:**

I *WOULD/WOULD NOT CONTRIBUTE TO/USE* **an information bank.**

I am *STARTING/RUNNING* **a** *GROUP/CENTER:* **Publish my address and phone number.**

There is a *GROUP/CLUB/CENTER* **called:**

Contact through: **Address:**

Zip: **Phone:**

I have been *SATISFIED/DISSATISFIED* **with service from this parts supplier:**

Name:

I know of the following *INFORMATION/SOURCE* **concerning:**

The details are:

LAST BUT NOT LEAST

We are going to try to list centers, clubs and sources of information. We also want to know of kits, products and designs that are not commonly known, and which parts suppliers do a good job - quickly supplying quality parts at a fair price. And about those who don't!

The good ones we will mention - such as M&R Enterprises, P.O.Box 1011, Sunnyvale, California. Marty Spergel is a regular at Homebrew Computer Club meetings - has to do a better job, or he doesn't get out of the building intact...

We plan a postcard sized file. If you decide to help, help us some more by either mailing us the form or sending info on postcards!

Evaluative Criteria for Remote Computer Services

DON RUSSELL

Director Information Systems
Department of Public Instruction
State of Wisconsin

This document is intended to serve the purpose of assisting school district personnel in evaluating the costs, benefits and support services to be obtained by acquiring instructional computing services from remotely located services via telephone transmission methods.

Quality of Service Provided

A. Computer response time

For school districts which seek to acquire computer services for use in their instructional program, the remote computer system accessed by telephone lines is becoming popular but care must be taken to assure that the system being proposed is capable of providing immediate response. This means from the time a student types in a request, the maximum time for the computer to respond should be either immediate or at most one or two seconds. Typically, there should be no noticeable delay in computer response. These remote computers are called "time shared computers" (several people in remote locations are sharing the machine at the same time) and they are capable of this kind of response.

Response time is extremely important in educational applications because of the need for immediate reinforcement of learning. It is especially important where the computer is being used to administer drill and practice, but other interactive programs also suffer when response time is poor. It can be extremely frustrating for a student to have to wait a long time for a response. Response times as high as ten seconds can be extremely wasteful of a student's time and thirty or more seconds can be intolerable. Cost effective utilization of equipment is also needlessly depressed by poor response times. There are suppliers who will minimize the significance of this feature but inevitably investigation will reveal inadequacies in their response times, hence their defensiveness in this regard.

B. Turn around time

Turn around time is the total time for completion of an entire assignment or problem. Such a task may involve several inquiries and corresponding "responses."

A good quality time shared system with rapid response also provides quick turn-around on student assignments as well as offering the student the opportunity to do all of his work at the terminal. This is one of the major advantages of timesharing over punched card computer systems. Card oriented computer systems (also referred to as batch processing systems) are ones in which students and other users submit a batch of cards containing programs and/or data. If errors are found in the programs and/or data, the student must return to the keypunch machine, make corrections and run the job again. Coupled with waiting time, the repeated trips to the machine can make the process a very frustrating one for students. Although batch processing still has a definite place in computing, it is becoming clear that timesharing offers a better all round instructional program and is actually superior for a host of instructional applications.

BASIC

C. Programming languages

A timesharing system to be used in school districts should offer a high quality programming language, preferably BASIC, and a modern CAI author language such as Coursewriter III or IDF (Instructional Dialogue Facility). BASIC is an acronym for Beginners All Purpose Symbolic Instructional Code, a computer programming language developed at Dartmouth College for use in educational. The selection of BASIC over any other single language is based upon its design criteria: It was designed to be (a) easy to learn and use, but (b) versatile and powerful enough for a wide range of instructional applications, and (c) oriented toward the educational user. A CAI author language facility is considered a must since this area is experiencing a considerable resurgence.

D. Commands to the computer

Whatever the language it should have simple, English-like, easy to use commands which allow students and teachers convenient access to the system. Commands are distinguished from the computer program itself. Commands generally tell the computer system what to do with the program being used. Unfortunately, too many computer systems require cryptic, highly specialized commands or a rather formal "job control language." A system to be used for instructional applications must keep these complexities to a minimum since such control languages reduce the accessibility of the system and frustrate teacher and student alike.

E. Language features

The programming language should be simple and easy to use. The high school instructional program is designed for all students, encompassing a wide range of ages and abilities, and an instructional program utilizing the computer should not, and need not, be aimed at an elite group of students, but rather at the entire student body and all curricular offerings. Since computers have become an integral part of our society, it is important that all students become familiar with the nature and operation of them. Experience has shown that some low achieving students responded quite enthusiastically to instruction on computers. More advanced students can certainly be motivated and accommodated with special programs, but it would be a grave error to overlook the documented successes which have been experienced with students of less than exceptional ability.

For the above reasons and others, it is strongly recommended that the BASIC language be available and this also explains why it is the most popular language used in education today. Any language for general student use should not be abstract. Another timesharing language, APL (A Programming Language), for instance, is quite powerful but is rather mathematical and abstract and is quite difficult for the average student to use, let alone to master. FORTRAN and COBOL are not particularly abstract, but are professional programming (and thus somewhat cryptic) languages designed for scientific computation and business data processing respectively. As a result, they are much more difficult for a student to learn and use as a probelm solving tool.

F. Additional languages

Additional languages are desirable but not necessary. A common argument states that languages such as FORTRAN and COBOL should be taught in the high school because these languages are commonly used in industry. However, these languages are intended for highly specialized applications and are used largely by professional programmers. The argument that they should be taught in high school is not a strong one since students are not likely to fill professional programmer positions after high school graduation. Computers have been common now for nearly 20 years and the job market in computers is much more competitive than ten to fifteen years ago. Those who seek a career in computing are now more likely to attend a vocational school or a four year college or university to acquire the necessary professional skills to become a programmer or even to function as an operator of these complex machines.

BASIC provides enough features to teach students those concepts of computing that will serve them later in their training and career. It is much more desirable to teach concepts in the high school, rather than to attempt to provide vocational training for programmers, which could become an extremely costly project and one without a marketable product in terms of a trained programmer or operator.

G. Problem solving facilities

The computer is an extremely useful tool in a wide variety of subject areas, not just in mathematics and science. The computer may be resented as an expensive toy if it appears to be the exclusive domain of the math department. Since use in other subject areas is to be encouraged, facilities for problem solving should be available. These facilities can range from a good BASIC language where students can write their own problem solving programs, to pre-written program that do tedious laboratory calculations.

The computer is a powerful tool which students in many subject areas can program to solve problems which would be difficult to solve by hand. Hand computation tends to be laborious and too often prevents a student from perceiving the concepts involved. Once the student learns to do the necessary calculations, it is often best to free him of this burden. For example, the computer has been programmed to do laboratory calculations for some courses and as a result students are able to perform more experiments with the same time investment. This enriches their educational experience, surely a worthy return on the investment.

H. Simulation programs

Computer simulation is a process whereby a teacher can use a role-playing "game" in which the students are able to "simulate" some real life process. As a result one can create a rather realistic laboratory experience where one could otherwise not exist. Simulation can be done without a computer, but usually the amount of hand computation required makes it impractical. Computer simulations are available in business, social studies, mathematics, biology, chemistry and physics to name just a few. Due to the wide subject range of these tools and the fact that they are usually used by teams of students, the use of the computer is extended to many more students than could actually sit at the terminal at one time, thus multiplying the student/machine ratio beyond the obvious (but inaccurate) one-to-one ratio usually perceived at first thought.

POLSYS — Political system simulation which is brutal to figure out but is exciting once you do. This simulation is designed to teach students how political decisions can be influenced by community action at the local government level. It is a rather modern simulation in keeping with the times and the 18 year old vote.

SLITS — An "extended Lab" experience for students who are learning about Young's Double-Slit experiment. Like the others, this program comes complete with a mini-text and lab guide.

A sizable package of simulation programs should be available and should encompass a wide variety of subject areas. The most noteworthy effort to date has been the Huntington II Computer Project, which was developed at State University of New York in Brooklyn under NSF sponsorship. Huntington II has produced a large number of high quality computer simulation games for use in secondary school classrooms. In addition to subject area enrichment, they offer a means of increasing student motivation and encouraging use of the computer by teachers outside mathematics. Numerous other simulations are also available and are invaluable tools for the classroom.

Any timesharing network proposing to serve the instructional user should have the entire package of Huntington II programs available in the system library, as well as many from other sources.

I. Computer Assisted Instruction and Computer Managed Instruction

When considering future needs, educators should consider expansion into computer assisted instruction (CAI) and computer managed instruction (CMI).

In "classical" CAI, the computer is used to either tutor or to administer drill and practice directly to a student. Such an application requires many hours of time at the terminal by a large number of students and, if not properly administered, could work to the detriment of a typical high school program. Yet when the terminal is not busy with other uses (which should have higher priority), it can be effectively used for some CAI applications. Summer use is probably a good example. A large number of drill and practice programs for use in elementary and remedial instruction are being collected and use of these applications particularly during summer and other "off-periods" is expected to grow considerably in the future.

In computer managed instruction (CMI) the student receives no direct contact with the computer. Instead, the computer is used to record his progress, to assist in proper student grouping, to relieve the teacher of some of the burden of record keeping and to improve the quality of the information recording and reporting process. Because individually guided education programs, such as Wisconsin Reading Design and Developing Mathematical Processes (DMP), impose massive record keeping and student grouping burdens on teachers, this educational use of the computer is expected to grow considerably in the future.

Rapid growth is expected in CAI and CMI in the near future, much, but by no means all of it, in the elementary school. The computer will offer teachers the capability of individualizing instruction much more than personnel resources have ever allowed in the past. In areas such as mathematics drill and practice, packages are already available that allow schools to give individual attention to a student that would have been prohibitive because of limited teacher time.

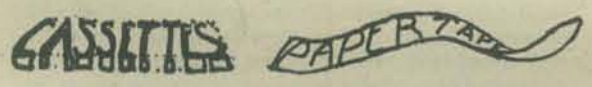
Computer managed instruction, although quite new, is growing very rapidly, and with computer costs coming down, it is probably only a matter of time before many schools have computer record keeping at least for individually guided education programs.

Although the initial thrust in CAI and CMI is expected in the elementary area, it is also expected to affect junior high schools and high schools since they typically must find some way for accommodating differences in background among their students and for assisitng in the remediation process where necessary. These applications are also assisting the process of individualized instruction and will address the same record keeping problems at these age levels.

Thus, timesharing networks should at least offer their participating schools the potential to expand into the areas of CAI and CMI at a reasonable cost some time in the future. A CAI author language as previously mentioned should be a highly desired feature of a proposed system.

J. Facility for programming

The system should provide the capability of teaching programming and computer science, in other words the ability to use the computer as an object of instruction. Instruction in computing and computer programming is an important and rapidly growing new subject area in high schools and colleges. The terminal offers a facility for students to write, debug and run their programs and enough capability to teach most of the basic concepts involved. Instructing students in programming without such a facility represents a nearly impossible task. It is a skill such as driving a car or riding a horse which can only be learned by doing. In addition the cultural impact of the computer can be most readily mastered by actual contact with a real system.



Features of the Computer System

A. System capability

The timesharing system used should be readily usable by the novice but should still be of sufficient sophistication to serve more advanced users also. Access to a medium sized timesharing system is therefore recommended over a stand alone mini-computer located in the school itself. The stand alone mini-computer capability is likely to be less versatile because of "program library" restrictions. Also, a timeshared computer is vastly superior in terms of program size, library availability and data storage size and is also more accessible to students and teachers than is a programmable calculator, which is often suggested as an alternative usually by the vendors of these devices.

B. Language support

A timesharing system used for secondary school applications should provide the following language features.

1. It should allow reasonably large programs, typically in excess of 500 lines of BASIC code. This will insure that the system is powerful enough to handle most of the programs which a high school teacher or student is likely to write or use.

2. The computer programming language should be versatile enough to allow teachers to teach basic computing concepts. As stated above, BASIC does meet this criteria. There are other languages that do also, but as stated previously they are not easy for high school students to use and learn.

C. Library facilities

The system should have library facilities. A program library is a package of "canned" programs kept on the computer's auxiliary memory such as a magnetic disc on a continuous basis and available to every user of the system. They can be accessed quickly and executed by users whenever a particular computation is required. A library of pre-written (or "canned") programs is particularly important because they help extend the use of the computer to a wide ragne of subject areas by people with little or no knowledge of programming.

There should be a main library from which all time-sharing users can fetch and retrieve programs. This library should be protected so that users cannot alter or remove the library programs stored therein.

Users should, however, have a private library reserved for their own school use where they are allowed to store their own programs and data. This private library is important because, if students or teachers use a particular program of their own often, they may need instant access. Reading that program into the computer from paper tape may take twenty minutes or more each time, whereas the recalling of it from a private library which is disc resident would be very fast.

D. Off-line storage needs

The timesharing system should support paper tape, magnetic tape cassettes or other off-line storage media. Even though it can be time consuming to use, paper tape is the cheapest and most popular off-line storage medium. A roll of it typically costs about $1.00 and can store hundreds of thousands of characters of information. As a result, hundreds of programs are usually stored off-line by teachers and students in a box or desk for only a few.dollars. In addition, since it is cheap, paper tape is disposable and is excellent for storing programs that are not yet in their final form. On the other hand, on-line storage such as library space is relatively costly and should be used judiciously.

Off-line storage media can also be used as a means of increasing the throughput of a terminal. If the school has an off-line terminal, students can use it to prepare their programs at human typing speeds and later enter them on-line to the computer at machine speeds.

Terminals and Communication Equipment

Other considerations which are important when looking for timeshared computing services include the terminals and transmission equipment which are used to communicate with the computer. A wide variety of these computer terminals are on the market today and attempting to select one can be a frustrating experience.

A. ASR-33 Teletype and others

The ASR-33 Teletype has been an accepted standard for some time. That unit has sometimes been compared to a Volkswagen; it is slow, noisy and ugly but cheap and dependable. Though it has some undesirable characteristics, the features it provides for its price make it the most popular timesharing terminal today.

An outright purchase of an ASR-33 typically costs about $1,000. Though rugged and dependable, it does require occasional servicing, and service contracts can often be obtained from a local supplier. Some TTY owners service their own units but this is not recommended unless there is a skilled technician on your staff. When the unit is leased from the telephone company or another vendor, service is usually provided as part of the lease arrangement. For beginners it is recommended that leasing of ASR-33 Teletypes with maintenance contract is preferred. The cost at present is $62 per month.

The ASR-33 Teletype prints at ten characters per second, about the speed of a slow reader or a fast typist. This is not the most desirable speed for some applications, but is adequate for most high schools in a computing program; also faster speeds can also exceed the speed of the reader and his ability to respond.

The ASR-33 also has a paper tape reader and punch provided at the base price mentioned above. This allows all the advantages of paper tape mentioned . On more expensive terminals these paper tape devices may cost $2,500 or more.

All of this is not to say that the ASR-33 is the only choice for a terminal. There are many finer units but typically the more features they offer, the more costly they become. For instance, those that are three times as fast are often three times more expensive and offer video displays but usually do not offer paper tape or a replacement for it except as an expensive option. Any unit selected for an instructional program should have at least as much capability as the ASR-33.

It is probably advisable to start a program using an ASR-33 and to supplement or replace it with more deluxe computer terminals that meet specialized needs.

B. Communication equipment.

Additional equipment is required to physically connect the terminal to the computer. There are a myriad of possibilities here also. In all cases, data sets (code translators) are needed at both the computer and the terminal ends of the communications line to code and encode the data signals and telephone lines link the data set at one end to the data set at the other end. Unfortunately, these transmission arrangements and costs vary widely depending on your location and distance from the computer, which makes it impossible to give a rule of thumb, but the costs can be readily computed and are of sufficient magnitude to warrant scrutiny. Some networks require the members to arrange and pay for their own communication costs individually, while others procure and pay for them for the members and charge them back as part of the fees. It is recommended that the method used be noted and the attendant costs be detailed for comparison purposes.

Some of the most common arrangements are:

1. Hard Wire. If users are within a few hundred feet of the computer, this technique generates the lowest communication cost.

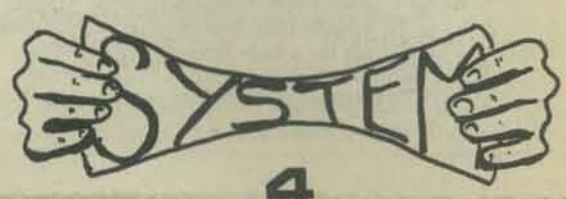
2. Dial Up. This technique requires the terminal user to "call" the computer over normal telephone lines. This technique is economical mostly when one is close to the computer or is within a toll free dialing distance of the computer such as a large metropolitan exchange.

3. Leased Line. When one is a long distance from the computer or when the local telephone exchange is not of sufficient quality for data transmission, leased lines are often the best alternative. With this arrangement, fixed phone lines and data sets are connected between the terminal and the computer. No dialing is required; to get a connection one usually just pushes a button. For this service, costs are calculated on a monthly mileage basis rather than the dial-up time tariffs with which most of us are familiar. For these longer distances, remarkable low total costs can result and it is a popular technique.

Typical communication costs should not exceed approximately $90 per month; this cost would include both data sets and inter-connecting lines. Costs, however, could either be much lower or much higher than that depending on your distance from the computer site. If you are within a reasonable distance from the computer, say 50 airline miles, you should be able to (a) lease a terminal, (b) provide maintenance, and (c) pay for communications (data sets and lines) for about $150 per month in addition to the charge for use of central site services and equipment. Often, very great distances may be involved without encountering increased costs by use of state "Telpak" lines (specially packaged low-cost lines). It is best to inqurie specifically about each case. The Wisconsin Department of Public Instruction will provide this assistance when requested.

Expandability of the System

Starting an instructional program in computing and placing a terminal in your school is only the beginning. If the program is successful, you likely will have to consider the long-range future of the program. It is quite likely that you will want additional services, including additional computer time and on-line storage as well as special programs and services in the future. Additional access time and on-line storage should be available at a reasonable and published cost. You may even reach the point where you want a second on-line terminal. All of these potential services and their attendant costs should be detailed and made available in advance.



Current Cost Estimate and Level and Timeliness of Service

A. Current cost estimate

For a total cost of approximately $4500 per year (12 months), a school should be able to pay all central and communications costs and receive a minimum of about 200 hours per month of computer connect time with the majority of that time occurring during the normal school day. For approximately $2500 per year (12 months) the equivalent of 75—100 hours per month should be possible to obtain. Some time sharing systems offer dial-in service on an hourly use basis and price quotations for this may be obtained.

B. Amount and timeliness of service

The object of acquiring timeshared computing is to make it available for your instructional program. Thus the timeliness and amount of on-line computer time is a most important consideration! Unless there is enough of it, the computer is not sufficiently available to students, thereby defeating the main purpose of the program. Access at least during a significant portion of the school day is also a mandatory requirement. Some networks have allowed their member schools ten or fifteen minutes computer time per hour during the school day. The rationale is that the off-line potential of the terminal will offset the on-line restrictions. The run-time error detection features are thus lost and the pedagogical significance is inordinately lowered since limited access is an irritant which actually discourages use of the computer.

It is imperative that a school receive a generous allo*cation of time for the money spent!* There are no clear standards, but some guidelines are as follows.

Central Site Support

A. Teacher education

The key element in any instructional computing program is the teacher. His or her preparation and leadership is the ingredient that will determine the success or failure at any given school. Those who are familiar with computers will often agree that its potential in instruction is limited only by the knowledge and imagination of the teachers using the system resources. Thus, any organization supporting instructional timesharing should provide the means for educating teachers or others who will use the service.

1. Short work shops — Orientation workshops in the use of computer terminals, often at user schools, should be conducted and these are usually sufficient to get a school started. Frequently, schools have one or more teachers who have worked with computers. They are often able to learn a system quite easily and provide the necessary leadership for their school's program. These sources are usually adequate for beginning a program, but to insure its ultimate success, more extensive training is desirable.

2. Graduate level courses — One of the most effective ways for teachers to improve their skills in this area is graduate level course work on the uses of computers in instruction. Such courses should be offered frequently at times and locations that are accessible to teachers and prerequisites should be minimal. "Computers in Education" courses should not be merely computer programming courses, but should give comprehensive treatment to the use of computers in education. Administrators should recommend that as many teachers as possible from the district take the course as leadership from teachers is far more desirable than mere technical knowledge of the computer.

3. In-service training — The school district can also help improve teacher skills by providing in-service training in educational computing. Often the entire faculty or teachers from selected subject areas can benefit from timely instruction in computer applications in their area. The network should assist in this task by providing a resource person to assist in the instruction and to make the materials available.

B. Personnel support

For those schools which decide to associate with a timesharing network or to acquire timesharing services from an educational institution, the amount of personnel support given by the central site is extremely important. The central site which provides computer time and services to school districts should be offering a full service rather than simply selling excess computing time. This practice, not at all uncommon, is proving to be largely unsatisfactory particularly for the novice schools in their initial effort. Computers are complex enough so that when problems arise, teachers and staff will have questions to ask. Also, when starting a new program, there should be assistance to the teachers directly and provision of inservice training.

1. Computer operators — The network center has the responsibility for running the computer, providing maintenance for it, loading programs and performing all the utility tasks that keep it running. In addition to the continuing responsibility to keep the system trouble free, there will undoubtedly be requests to make of the people at the central site. Thus the network should be able to provide assistance incldiuing computer operators to handle routine requests, to tend the system and to take trouble calls.

2. Programmers — Assistance should also include professional level staff to write general purpose programs develop new programs and publications for the network, and to acquire materials from other sources. Computing instruction is expanding at a rapid rate throughout the country and much of the material produced by other computer projects can be useful to your program. Someone has to keep an eye on this field, to acquire the materials and if they are suitable for use to adapt them to the local system and make them available in general.

3. Resource persons — The network should also have a user relations coordinator to provide direct on-site assistance. This person should be available to give faculty and staff an initial orientation, and in general, to assist in training teachers. Also, he/she should be available for specialized workshops for groups of teachers in a specific study area. Finally, this person should act as liaison and facilitate communication. The cost for such services, if any, should also be clearly detailed.

C. Communication between users

Part of network operations should include good communications. The network should supply at least an initial set of manuals to its subscribers. These include both the computer manufacturer's reference manuals and any manuals that the network itself provides.

To keep its users informed, and to foster closer cooperation among the users themselves, a local network newsletter is a very worthwhile vehicle. Typically, network members can also subscribe to the computer manufacturer's newsletter for no charge. Both devices are invaluable in keeping posted on new programs, literature and teaching materials.

Meetings should also be held where teachers and others can gather periodically in order to exchange ideas, problems, information and programs. It would be to their advantage if the netowrk users were allowed and encouraged to organize an informal organization and to actively support it.

Vendor Stability and Commitment to Program Support

A statement of commitment to a full program of support and services should be available from a high level administrative officer of the organization which offers timeshared computer services. Generally speaking this commitment should emanate from the level of university chancellors or vice chancellors, CESA coordinators, etc. The stability of long range intentions is a factor in the decision process for the school district which wishes to evaluate a potential source of service.

computer nut

Between the stages of pre-calculation and scientific speculation you are bound to find yourself confronted by a Computer-nut.

Usually you can spot one a mile away, merely look for someone wearing sneakers and carrying a briefcase or deeply involved in writing a program on the backside of a flattened dixie cup.

But spotting a computer-nut is not easy for there are several varieties.

First there is the all-out total know-it-all programmer, he is the one with the buzz haircut and horn-rimmed glasses.

Then we find the computer fanatic, he feels that good manners should be used in the computer room at all times (so he can use the computer undisturbed) and force should be dispensed with, but force is the only thing that will get him off the terminal.

Then of course there is the honest computer nut, he is the one who feels guilty every time he breaks off a program.

Last, our No. 1 enemy, the computer-nut that puts in repeating joke programs with disabled break keys.

Whether you are one of these or not it does not matter for you are still a new breed clearly classified as a Computer-nut.

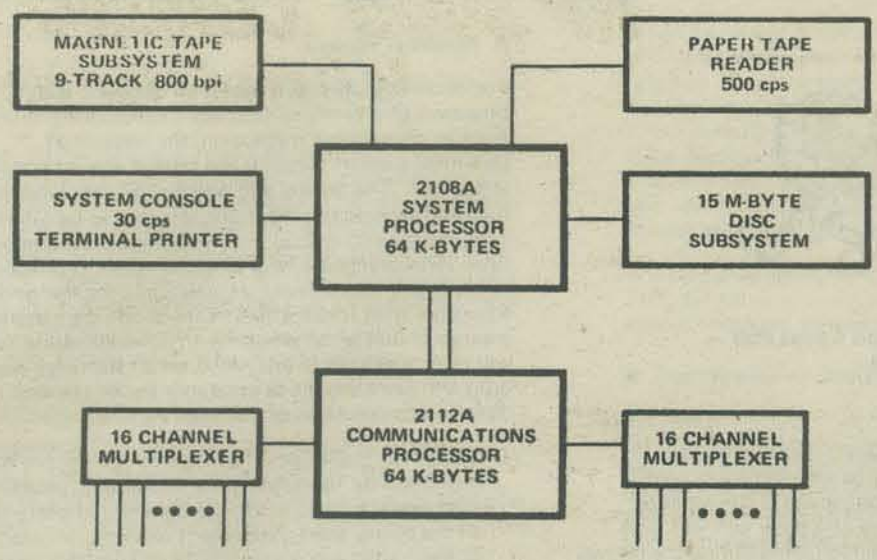
Randall Galliher
5910 White Cloud
San Antonio, TX 78238

1. Larger string capacity and additional string operations.

2. Facilities for the creation, deletion and manipulation of data files from a program.

3. Facilities for program access of peripheral devices such as magnetic tape, line printers and card readers.

4. Access to operating system services under program control.



HP 2000 ACCESS SYSTEM MODEL 40

You've heard about the new HP2000G? That's what everyone thought it would be called. Upon release, they changed the name to *HP 2000 ACCESS SYSTEM.* It's an improvement over the 2000F in three ways –

- It can act as a remote job entry terminal to any IBM system operating in HASP or a CDC system in their U200 concurrently as time share users run BASIC programs.
- It allows all users access to peripheral resources; card reader, line printer, paper tape punch, paper tape reader, magnetic tape unit.
- It has some nice enhancements to the BASIC language.

The HP2000/30 is a 16 terminal system with two HP2100 processors, a 32K system for CPU and a 16K system for communications. For $59,960 it comes with a 5M byte disk, magnetic tape unit, and paper tape photo reader. This configuration will not operate on an RJE terminal without additional storage. in the processor unit.

HP2000/40 is the 32 terminal system which has 2–32K processors. It comes with a new 15M byte disk plus magnetic tape and paper tape reader. Yours for $67,600. Software to operate the RJE feature costs an additional $1000. (Upgrade *costs* from 2000F vary from $8500 to $11,500 depending on what configuration you have)

The BASIC language improvements include –

*UPS$ function replaces lower-case letters with their upper-case equivalents.

*POS function allows the scanning of a string for a particular substring, e.g. great for searching for a key word in a string.

*Number of string variables increased from 26 to 78 and the maximum length of each variable increased from 72 to 255 characters.

*CHR$ function returns a one character ASCII string when given the decimal number of the character. Any characters, even control ones, can be generated.

*CONVERT statement converts a string to a unique floating point number and vice-versa.

*NUM function returns a decimal number (0-255) equivalent to a supplied ASCII character. This is the reverse operation of CHR$ and allows the program to compare or branch on the value of a particular character.

*EXTENDED STRING LITERALS allows users to enter special ASCII control characters into a BASIC statement, e.g. output a carriage return to the terminal through the PRINT statement.

*CREATE and PURGE statements.

*FILE Command associates a file name with a peripheral device.

*ADVANCE statement allows skipping a number of items in a file.

*ASSIGN statement allows dynamic allocation of device in running program.

*UPDATE statement allows updating of a specific item in a record.

*REC and ITM allow determination of file pointer location by record and item number.

*LOCK/UNLOCK statements allow multiple users to cooperate in multiple write access to a file.

*READ, LINPUT and PRINT statements now allow accessing of peripheral devices.

*SYSTEM statement allows variety of operating system commands to be issued from a program.

OLD NEWS – The HP2000E 16 terminal system with 5M byte disk still runs $34,950. The 2000F, 32 terminal system with 5M byte disk now runs $63,500.

Reprinted with permission from
Hewlett Packard
11000 Wolfe Road
Cupertino, Ca. 95014

HP 3000 COMING OF AGE

Is it ready for you?

It's hard to write about this system for any number of reasons –

1. Users are few, and finding *hard-core* information about the HP3000 is difficult.
2. HP literature, while clearly defining the hardware and software, does little to discuss terminal responses, disk vs core vs response time pay offs, etc.

With that disclaimer we proceed. This article is written based on heresay. We felt it absolutely essential to introduce the HP3000 as "its time has come" and its impact may be great in the "ed biz." None-the-less, we have many, many questions. If you have experience with the 3000, please, please write us a publishable letter describing your experiences.

The HP 3000 was announced some 3 years ago. It was hailed as the first multi-language, time-sharing system at mini computer prices! Time sharing and batch would run simultaneously with service to as many as 64 users. A new style architecture would support all this in just 96K bytes! From then on it was straight downhill. HP had a hell of a time delivering systems, let alone promises. After a year and a half of false starts, unhappy customers, and declining profits, the company essentially pulled the product off the market while the factory retooled, retrenched and reevaluated their product. Early systems would drive few terminals and they ran slowly. Any *multi-language usage* ground the system to a slow drudge 64 terminals was a fantasy, 32 a dream. The reality was 12–15 running slowly with, not 98K bytes, but 144 bytes.

The HP3000 has returned! And this time it seems to be real and here to stay. HP has developed 4 configurations each a "building block" expansion of a smaller configuration.

HP3000 Model 50CX

The smallest unit "for low cost terminal capability." A 96K byte system with 5M byte disc and magnetic tape unit. A 16 channel multiplexer is included for $99,500 total. Free software for this model is limited to a text editor (EDIT), HP's new compiler language SPL and some utilities. The system will run 4–5 terminals depending on the job mix.

Model 100CX

For $129,500 you get a more complete and more capable system. Still 96K bytes of memory, the Model 100 has a 15M byte disk, magnetic tape unit, card reader (600 CPM) and line printer (200 LPM) In the minimum configuration it will "support 4 to 8 terminals. With optional core expansion it can support as many as 16 terminals." You get the Model 50 software plus BASIC interpreter and BASIC compiler as part of a time share option for FREE. For $5000 you receive a Scientific Software Package which includes a FORTRAN compiler, scientific library, statistics package and other goodies. Presumably these can run simultaneously with other software. For another $5000 ($9000 for the two) comes an RPGII compiler plus goodies. Looks like a pretty complete system!

Model 200CX

This is "big brother" with bigger price, $171,000. It's 128K bytes is coupled with a 15M byte swapping disk to speed up responses plus a 47M byte moving head disk. To all the other goodies from Model 100, you can add a COBOL compiler as part of the $6000 Business package. This configuration *should* run 16 terminals with suggestions that more are possible.

Model 300CX

The top of the line features terminal and batch users with as many as 32 users. It's cheap for what it does when compared with what other vendors would require for BASIC, FORTRAN, RPG, COBOL, and a Data Base Management System, all operating at the same time. $203,500 which includes 128K bytes of memory, 15M byte disk, 47M byte disk, 1250 LPM printer and a card reader/punch system. Model 300CX is a 200CX with faster peripheral devices and a free data base management software system.

All HP3000's will run foreground and background jobs simultaneously – time queues can be set up on equal or unequal priority for 3 user types –

1 – timesharing
2 – batch
3 – "slop" . . . do it when you can!

The preceding 4 models are "packages" – actually, all software runs on all systems. You can run timeshare BASIC on the Model 50, however, slowly. You can run data base on the Model 200, etc. These packages are probably make *efficient* use of hardware. You must pay $$ for software that doesn't come with your model.

There is much much more to the HP3000 than these descriptions can offer. There are many payoffs keyed to your specific needs. HP now has software to do school or college student recordkeeping and budget operations. If that's your application then your configuration will be far different than that of a school devoting its system to student use only.

If you look at the 3000, move with caution, ask lots of questions, **talk to people who own one** and particularly talk to someone who's use is like yours. The HP3000 seems to have arrived but it's not as easy to buy as a simple BASIC timesharing system.

The HP 3000: What It Is & What It Ain't for Education

For educational application, the Palo Alto Unified School District has an HP-2000 and an HP-3000. The 2000 is used in elementary and secondary schools for BASIC programming, simulations in social studies and science, educational games, guidance counseling, and computer assisted instruction in language arts, math, foreign language and other subjects. The 3000 is used in the high schools for instruction in BASIC, FORTRAN, COBOL, SPL, and computer science. It is used primarily by the Math and Business Education departments, although it is also used for educational research and testing. The district will soon be using the 3000 for data processing, but more about that aspect later.

The Palo Alto schools' HP-3000 has 64K words of memory, an ISS disc of approximately 50 million bytes, a 7900A system disc, one 800 BPI tape drive, a 7260 optical mark card reader (run through a terminal), and a Tally 200 LPM printer.

The district is fortunate in having the two computers because neither one alone would satisfy the needs of the district, even if either one were able to support the 48 terminals now in use. The 2000 is ideal for CAI because of its fast response time, but it is limited to BASIC only. It can only do batch processing when the time sharing system is shut down and a different disc operating system is loaded. To do so is time consuming and either eliminates all other users or must be done at night – an awkward situation in terms of personnel shift hours. As the computer center now operates, the HP-2000 is up on line with 32 ports 24 hours a day except for about one hour daily for backup. The 2000 has a surprising number of evening users – teachers doing their preparation, students doing CAI at the Children's Hospital, for example.

The HP-2000 has a relatively simple operating system. Using only the 2000 a student would learn little about computer shops using bigger and more complex computers. Conversely, the HP-3000, although a relatively inexpensive mini-computer, has a big computer-type architecture with a big computer-type operating system. Students trained as programmers or system operators on the 3000 make an easy transition to bigger systems. They learn to use sub-systems such as the text editor and the file copier. They learn other languages such as FORTRAN and COBOL that equip them better for jobs in certain applications.

The HP-3000 system is flexible; time sharing and batch processing can operate simultaneously, *but* there are trade-offs, given the fact that a mini-computer is trying to pretend that it's a big computer. Response time with only a few users is slow. CAI is an entirely inappropriate activity for the 3000. It can easily be overloaded, especially when compiling programs; with more than 12 interactive sessions, the machine slows down to a crawl.

Additional core and a fixed-head disc would improve performance to a marked degree. It is reported that without additional core, a fixed-head disc improves the efficiency from 20% to 30%. Additional core should bring the response time to a point comparable to the 2000.

As it is, considerable attention must be paid to tuning the performance to the needs of a particular installation. As the district faces bringing its data processing from a regional computing agency to the HP-3000, the district computer center has been and is experimenting extensively to refine that tuning. Obviously, the data processing must be scheduled to avoid impacting the educational applications. In addition, setting the quantum (the time slice given each time sharer) and relegating batch processing to a secondary queue improves the interactive response time. **The most effective tuning procedure is to spool all compiles, permitting them only to come out singly or at a time when the interactive load is low.** The district will be acquiring a new, faster moving head disc, which will enable the HP-3000 to support 20 ports with a response time at least no worse than it is now.

For the educational requirements the Palo Alto Unified School District makes of its HP-3000, it serves well. If it were the district's only computer, it could not meet the educational needs of the district. If such were the case, the HP-2000 would serve better as the sole computer, recognizing that it, too has its limitation.

Ernie Pope
Educational Technology
Palo Alto Unified School District
Palo Alto, Ca. 94306

IBM Wants You

by LeRoy Finkel

You may not have noticed but for 2½ years we've been writing about classroom computer hardware with nary a mention of the world's largest hardware manufacturer, IBM. Why? Because for 2½ years we didn't feel IBM had much to offer the classroom computer user, either in hardware or software.

This article represents a significant departure from past procedure but not a change in attitude. If your school, district, county, neighbor, friend, consortuim, etc., has an IBM 370 Model 125, 135, 145 or bigger, read on. This article will call to your attention what you can do to it to make it usable in your classroom. If there are no computers near you, but you're looking for an instructional computer system, go call your friendly DEC or HP salesman. . . they still have the best buy for you.

In talking with IBM we discovered two things –

1. Big computers use a whole new jargon which required a whole new learning process for us –
2. Simple, straight answers are hard to come by since there are a jillion different configurations for each IBM system.

IBM is unbundled and does not offer any time-sharing software for its systems – at least not down where your $ can afford it. They do recommend two sources to select from: WESTCHESTER BOCES of New York and McGill University, Canada.

BOCES BASIC

If you only need to talk BASIC and COURSEWRITER and you're near a 370/125 or larger, BOCES has a software package that may just be what you need. In COURSEWRITER they have written a BASIC translator that makes the user think she (he) is sitting at any normal time sharing terminal connected to any popular system. It provides line editing, error messages, plus all the language capabilities you are accustomed to. Our analysis of the BOCES BASIC language is that it looks like they took all the features of HP 2000 BASIC (good and bad), added a few APL functions plus a few other nice features. Programs are limited to 1000 statements with CHAIN available. They are continually upgrading the language and making additions and improvements.

What hardware do you need for BOCES BASIC? You need at least a 370/125 with 120K core plus some core for system overhead and 10–20 Mbytes of disk. If you want to run batch jobs in the background while time share runs in the foreground you'll need more core and more disk. One local district, San Mateo High School District, runs 14 TTY terminals in BOCES BASIC while running very small accounting jobs in the background. After school hours, San Mateo runs its regular school data processing jobs in the large partition that was used for time sharing. San Mateo estimates that the hardware required to run this time share system costs them roughly $30-35,000 more per year than the hardware necessary to run a straight batch DP shop. (Compare that cost to a stand alone system.)

BOCES BASIC has no real limit as to the number of terminals except core space and speed. San Mateo's 14 terminals run on a very small system. Response is alow when compared to an HP 2000 system but average when compared with other systems. The system is hampered by the fact that BASIC is simulated in COURSEWRITER requiring much system overhead and slow response times. Rumor indicates that BOCES BASIC will soon be available in machine code, reducing core requirements and increasing speed.

Costs – The BOCES BASIC software costs $1050 per year which sounds very reasonable. Add a $220 per month COURSEWRITER software fee and your total annual software charge is up to $3690. Buyer Beware – this software system is **not** supported by IBM. Software support comes from Westchester BOCES. They have done a nice job of cleaning up past bugs.

For more information –

Board of Cooperative Educational Services
Westchester County
Elmsford, New York

or

IBM

MUSIC

MUSIC – McGill University System for Interactive Computing.

We don't normally write about systems we haven't used but MUSIC seemed important for this article even though we haven't been able to arrange a hands-on demonstration.

Those of you who are accustomed to large scale time sharing systems will like MUSIC. It runs on nearly any IBM 360 or 370. A 240K system with 3–2314 disk drives or 2–3330 drives will support as many as 60 terminals. (384K will support 250 users.) Users can program in their choice of FORTRAN IV, ANS COBOL, Assembler F, BASIC, APL, Basic assembler and Basic FORTRAN. From what we can see the software "system" is as complete and thorough as you could possibly imagine. It includes a large Public Library of popular programs and various application packages developed by McGill or adopted from well known public domain packages.

A nice feature of MUSIC is that it can be one of many "jobs" running on your computer. Batch jobs could be running in the background while interactive users operate in the fore ground. Common files may be used by both batch and interactive users which opens up all types of interesting possibilities.

Our COBOL experts don't like the interactive COBOL because only fixed length sequential files can be used. However, for a teaching language or for small jobs, it looked good.

MUSIC BASIC looks like a full blown implementation of the language. The only serious defect we could see is that string variables are limited to 7 alphanumeric characters (subsequent software releases may have changed this). BASIC works with a compiler, not an interpreter or translator. That means that though you can call it "interactive" it doesn't 'interact' the way most time share users expect. To enter a program you follow these 3 steps:

1. Enter program using BASIC editor
2. Compile and check for errors
3. Execute compiled program

If errors are detected at step 2 or 3 you must return to the editor, change your program and then proceed to steps 2 and 3 again. Sounds easy but it does add some inconvenience for the new comer or occasional user who has to remember a more complex command syntax to get operational.

We've hear the following comments about MUSIC from users and friends of users –

- Great system. Teachers love it.
- With 30 terminals on a 370/135, it's really quite slow for students. They wait after they press RETURN!
- System software is solid.
- Installing the system is quick and clean.

When you ask IBM about MUSIC, they talk about systems in the 370/145 or bigger range even though the software does run on smaller systems. School users of MUSIC include Chicago School Board, Boston University, American University (Washington, DC) Boulder Valley Schools (Colorado) and Huntington Beach School District (California).

Costs – Buy MUSIC from McGill University and it will cost you $7000 to install and they send staff if you have difficulty bringing up the system. Annual charges thereafter are not clearly specified. Buy MUSIC from IBM and it costs $14000 and they provide "Level B" support (whatever that means). Hardware costs are nearly impossible to calculate. There are many, many payoffs to consider – most dealing with response time.

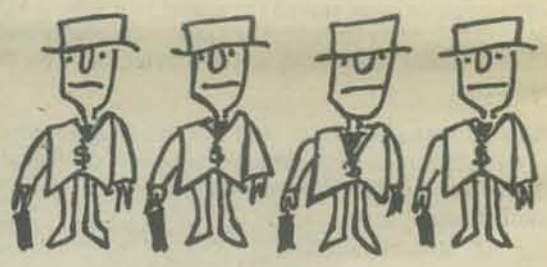
Some payoff questions include – will the system also be used for batch work?, how many terminals will be used? (MUSIC supports 60 – 250) how important is multiple language capability? To be sure, annual hardware charges for a MUSIC system will start at $150,000. The real question is, how much has to be added to an existing system to make MUSIC operational and that answer varies from system to system.

MUSIC is an alternative to consider. For more info contact –

McGill University
Montreal, Quebic
Canada

or

IBM



ALTERNATIVES

In an effort to compete with IBM's time shared offerings, DEC and HP have introduced hardware that will "talk" to most any IBM system. What they have done is made improvements to their regular time share hardware line to allow one channel to communicate to the IBM system on a remote job entry (RJE) basis. Any time-share user using the regular DEC or HP time share system can "que" his job so that at some point the information will be transmitted to the host IBM system. For those of you who talk in "IBM", the time share system emulates an IBM 2780 Data Transmission Terminal while still giving you all the regular time share capabilities.

Our local DEC representative priced an RSTS/2780 using a PDP 11/40 with 64K (words) core and 7.5M bytes of disk storage plus magnetic tape unit, RSTS software, 2780 hardware and software, at $86,000. This 32 terminal (31 time sharing, 1 RJE to IBM) system will cost $20640 or so per year on a lease purchase (you own it after 4-5 years) plus $8400 per year for maintenance.

HP's equivalent is the new HP2000 access system (see more on page 6). The model 40 drives 32 terminals in BASIC, more than one of which may be running RJE to an IBM system running in HASP operating system. Model 40 is $67,600 with 64K (words) of semiconductor, 15M bytes of disk, magnetic tape plus $1000 for all the software to do the RJE for a total of $68,600. Regular 5 year lease is $17,652/yr. (there may be an educational discount) plus $5856 per year for maintenance.

Classic Classic Classic

What's new from DEC—

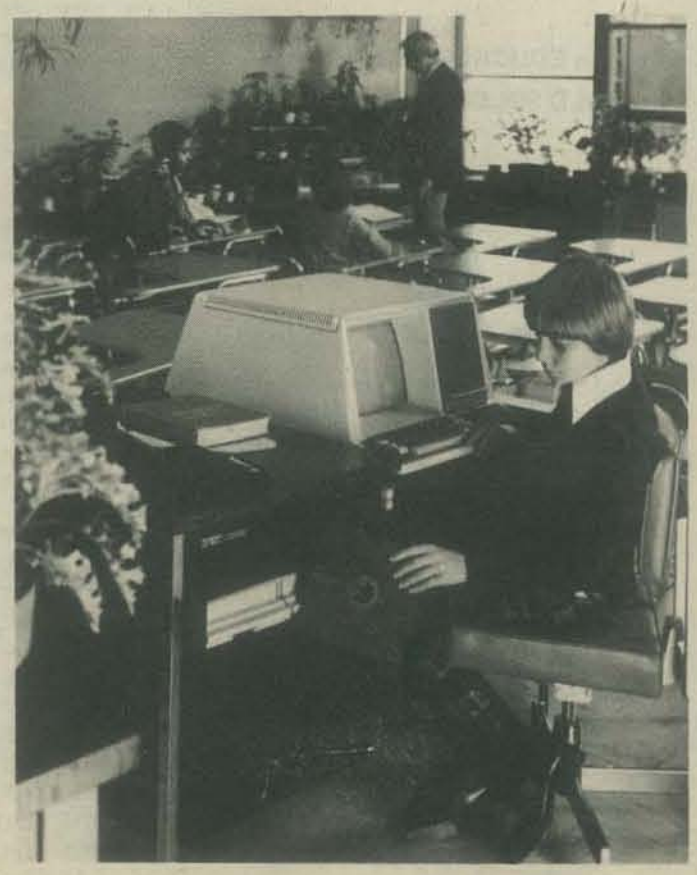
It's small. It's cheap (inexpensive). It comes standard with a CRT display terminal and printer. It includes dual "floppy" disks that hold 256,000 characters each. It talks in OS/8 BASIC. It's fast. It is sweet. It may even jump over tall buildings . . .

It's DEC's new CLASSIC (CLASSroom Interactive Computer). Built around a 32K PDP 8/E with dual floppys and a VT50 CRT/printer, this new system was designed with educators in mind. It's priced at $8900 or less than $200 per month on a lease. It weighs 150 lbs and DEC suggests that you roll it around from room to room!

The floppy disk makes it possible for each student to have his own cheap library of programs to carry around and easily plug into the system. With a little imagination, students could have their own operating system on a disk as well. DEC also provides a complete library of programs on floppy's — games, Huntington programs and regular public library materials.

DEC shipped a prototype CLASSIC around to sales offices for demonstration purposes. Our demo model was really beat up but we did get a feel for it. OS/8 BASIC is a BASIC compiler system. You do not get line by line error messages as you enter programs. You load, compile, run and then see your bugs. Those of you who are used to interactive BASIC interpreter will have to adjust to this system. But its so fast, the adjustment won't bother you.

The CLASSIC offers the educator some exciting new possibilities. If you're shopping for a single terminal system you must see this one (expansion to multi terminals is not possible... yet!). Of interest to you vocational educators, CLASSIC is being sold to businesses under the name Datasystem 310 for substantially more money. Also to engineers. Maybe you can justify its purchase for vocational purposes.



OMSI-RT Adds Multiple Languages to RSTS/E

ANSI FORTRAN IV and MACRO-11 Assembler

RELIABLE, POWERFUL, FAST, AND FLEXIBLE – COST IS $2350

FEATURES –

- RSTS/E users may enter RT-11 system to compile and run FORTRAN IV programs, assemble and run MACRO-11 programs, and use other PDP-11 languages.
- All users can switch at will between BASIC-Plus and RT-11 – all features of BASIC-Plus are retained.
- Standard DEC FORTRAN IV and MACRO-11 run unmodified under OMSI-RT.
- RT-11 computation speed can be 2 to 50 times that of BASIC-Plus.
- FORTRAN can read and write BASIC-Plus compatible files.
- Many advanced RT-11 features available including chain and overlay.
- System and other users are fully protected from RT-11 user errors.
- 28K word RT-11 jobs are possible depending on system capacity.
- RT adds only 4K words to resident RSTS code.

Strong Unconfirmed Rumors –

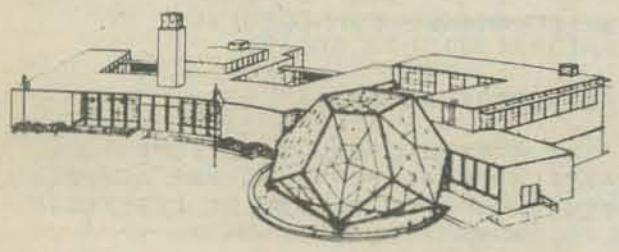
After two years of effort, the state colleges in California will award contracts for 19 interactive BASIC computer systems. Winning bidder is DEC RSTS/E on a PDP 11/45 mainframe.

LIMITATIONS –

- The RT-11 system requires more sophisticated users than BASIC-Plus.
- RT-11 real time and foreground/background are not available.
- Programs that do not use standard RT-11 I/O must be modified.
- Disks are the only file structured devices accessible under RT-11; other RSTS devices may be accessed as non-file structured devices under RT-11 or with BASIC-Plus utilities.

TRY IT NOW!

Call (503) 248-5900 or write for a demo account. OMSI's computer telephone is (503) 248-5961 available weekdays 8 a.m. to 6 p.m. After signing in, type "HELP RT11" for further instructions.



TO GET OMSI-RT –

You must have at least 48K words of memory; 64K is much better. Obtain the DEC binary licenses, software, and manuals for RSTS/E (version 5B or later) and RT-11 (include RT/FORTRAN if FORTRAN is desired).

OREGON MUSEUM OF SCIENCE AND INDUSTRY, 4015 S.W. CANYON ROAD, PORTLAND, OREGON 97221 (503) 248-5900

BIOSIN for a tiny EDUSYSTEM

- written in EDUCRUNCH BASIC
- by the OLD SOLDIER

Dear Dragonkeeper:

Here is my latest, best and hopefully last version of crunched BIOSIN, adapted to EDU20-C BASIC for an ancient PDP-8L with only 8K words of memory.

A word of caution. This would just barely fit in my 8K. As it was, most of the REM statements and explanation had to go. For these, refer to the Lawrence Hall of Science version in your earlier issues (PCC volume 3, number 2, pages 6 and 7.) Any brave programmer who wants to expand on them in 8K will have quite a problem.

Believe this should run on 8K Altair BASIC if backslash (\) is changed to colon (:) and LINPUT is changed to INPUT. Could perhaps run in 4K Altair BASIC if you could work it without strings, but that would be a job.

I would like to make one suggestion for others who submit programs. Please, even if you have a "super extended" BASIC, try to avoid using those functions not available to older and smaller machines except when they are absolutely essential to the execution of the program. Only in this way can we make our games and such, as widely available and useable as you would no doubt like.

On another point – I have been (I hope) substantially increasing my clout with public officials through the use of my computer. Almost any configuration can be easily programmed to write duplicate letters to multiple addresses. Business does it all the time. Whenever some action of our government grieves me, and this is quite often, I write one good letter, then plug in every public official remotely connected with the matter and let fly. Then I have a stiff drink or so.

Also, if anyone out there has a cheeep DECtape with control, out there in dragonland, let's haggle.

Sincerely,
The O.S.†

† old soldier

```
  1 REM "BIOSIN" BY LAWRENCE HALL OF SCIENCE, U OF CAL BERKELEY
  2 REM MODIFIED BY THE OLD SOLDIER FOR PDP8L, EDU20
 33 PRI"FOR WHOM IS THIS CHART?";\LINPUT Z$\Z=ASC(Z$)
106 FOR I=1TO7\READA$(I)\NEXTI
107 FOR I=1TO12\READA(I)\NEXTI
108 FORI=1TO12\READB$(I)\NEXTI
113 DEF FNY(Y)=29-SGN(Y/4-INT(Y/4))
115 DEF FNS(S)=INT(20*SIN((S/P-INT(S/P))*6.28318)+40.5)
117 DEF FNM(X)=INT(7*(X/7-INT(X/7))+.5)
159 PRINT " BIRTHDAY (MM,DD,YYYY)";\INPUTM,D,Y
163 Y=INT(100.1*(Y/100-INT(Y/100)))\IFM=1THEN174
165 A(2)=FNY(Y)
169 FOR I=1 TO M-1\D=D+A(I)\NEXT I
174 PRINT " PLOT STARTING DATE? (MM,DD,YYYY)";\INPUTM1,D1,Y1\DO=D1
177 Y1=INT(100.1*(Y1/100-INT(Y1/100)))
178 A(2)=FNY(Y1)
179 IF M1=1 THEN 189
181 FOR I=1 TO M1-1\D1=D1+A(I)\NEXT I
189 S=0\FOR I=Y+1 TO Y1-1\IF I/4=INT(I/4) THEN S=S+1\NEXT I
193 IF Y/4=INT(Y/4)  THEN S=S+1\IF D>59 THEN 197
195 S=S+((Y1-Y-1)*365)\S=S+(365-D)\S=S+D1
196 GOT200
197 IF Y/4=INT(Y/4) THEN S=S-1\ GOT195
200 D3=Y1*365+D1
203 FOR I = 1 TO Y1-1\IF I/4=INT(I/4)THEN D3=D3+1\NEXT I
206 PRINT" HOW MANY DAYS DO YOU WANT PLOTTED";\INPUTD9
209 D1=DO\GO SUB 300
211 PRI\FORI=1TO24\PRI"(*)";\NEXTI
213 PRINT"";TAB(30);"(-)      (0)      (+)"
214 FOR O=1 TO D9\E=FNM(D3-1)+1\PRI A$(E);\PRI D1;\IF Z<1THEN224
223 IF D1<> 1 THEN 226
224 PRINTB$(M1);\PRIY1;\Z=Z+1
226 IFA$(E)="SUN"THEN229
227 FORI=LEN(C$)+1 TO 63\C$(I,I)=" "\NEXTI
228 GOTO 230
229 FOR I=LEN(C$)+1 TO 63\C$(I,I)="-"\NEXT I
230 FOR I=1TO3\P=18+(5*I)\X(I)=FNS(S)\NEXTI
236 C$(40,40)="I"
237 C$(X(1),X(1))="P"
238 IF X(1)<>X(2) THEN241\IF X(1)<>X(3) THEN241\IF X(2)<>X(3) THEN241
239 C$(X(1),X(1))="*"\GOT251
241 IF X(1)<>X(2) THEN 244
242 C$(X(3),X(3))="C"\GOT245
244 IF X(2)<>X(3) THEN 247
245 C$(X(2),X(2))="*"\GOT 251
247 C$(X(2),X(2))="S"
248 IF X(1)=X(3) THEN 239
249 C$(X(3),X(3))="C"
251 PRITAB(14);\FORX=15TO63\PRIC$(X,X);\NEXT X
255 PRINT\D3=D3+1\S=S+1\D1=D1+1\IFD1<A(M1)+1THEN267
261 D1=1\M1=M1+1\IFM1<13THEN267
264 M1=1\Y1=Y1+1\A(2)=FNY(Y)
267 NEXT O
268 PRINT""
270 PRI\FOR I=1 TO 24\PRI"<*>";\NEXT I
272 FOR I=1 TO 10\ PRI\ NEXT I
277 DATA"MON","TUE","WED","THU","FRI","SAT","SUN"
278 DATA31,28,31,30,31,30,31,31,30,31,30,31
279 DATA "JAN","FEB","MAR","APR","MAY","JUN"
280 DATA "JUL","AUG","SEP","OCT","NOV","DEC"
281 END
300 FOR I = 1 TO 10\ PRI \ NEXT I
302 FOR I = 1 TO 72 \ PRI"*";\ NEXT I
304 FOR I=1TO5\PRI"B I O R H Y T H M    C H A R T";CHR$(13);\NEXTI
310 PRI\FORI=1TO4\PRITAB(4)"FOR";\NEXTI
311 PRI
312 FORI=1TO5\FORJ=1TOINT(Z/6+.9)\PRIZ$(J);\NEXTJ
313 PRICHR$(13);\NEXTI
314 Z=0
320 PRI\PRI\PRI
322 PRITAB(10);
326 PRI" YOU HAVE LIVED "S" DAYS AT THE START OF THIS PLOT."\PRI
328 PRITAB(10)CHR$(34);"P";CHR$(34);" STANDS FOR PHYSICAL CYCLE.";
329 PRITAB(57)"(23 DAY)"
330 PRITAB(10)CHR$(34);"S"CHR$(34);" STANDS FOR SENSITIVITY CYCLE.";
331 PRITAB(57)"(28 DAY)"
332 PRITAB(10)CHR$(34);"C";CHR$(34);" STANDS FOR COGNITIVE ";
334 PRI"(INTELLECTUAL) CYCLE.";\PRITAB(57)"(33 DAY)"
340 PRI
344 PRI"CRITICAL DAYS"CHR$(13);\PRI"CRITICAL DAYS:"
346 PRI
348 PRI"WHENEVER A CYCLE CROSSES THE MEDIAN LINE, THIS IS";
350 PRI" A ";CHR$(34)"CRITICAL DAY";CHR$(34);"," YOU ARE SUPPOSED ";
352 PRI"TO BE MORE PRONE TO ACCIDENTS ON THAT DAY SO BE CAREFUL!"
362 PRI\PRI\RETURN
```

```
***************************************************************************
B I O R H Y T H M   C H A R T
   FOR
T H E   P. C. C.   D R A G O N


           YOU HAVE LIVED  16583  DAYS AT THE START OF THIS PLOT.

          "P" STANDS FOR PHYSICAL CYCLE.                        (23 DAY)
          "S" STANDS FOR SENSITIVITY CYCLE.                     (28 DAY)
          "C" STANDS FOR COGNITIVE (INTELLECTUAL) CYCLE.        (33 DAY)

CRITICAL DAYS:

WHENEVER A CYCLE CROSSES THE MEDIAN LINE, THIS IS A "CRITICAL DAY", YOU
ARE SUPPOSED TO BE MORE PRONE TO ACCIDENTS ON THAT DAY SO BE CAREFUL!


(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)
                              (-)     (0)     (+)
TUE 15  JUL 75                       C P                    S
WED 16                           C     I     P             S
THU 17                        C        I          P       S
FRI 18                     C           I               PS
SAT 19                  C              I            S     P
SUN 20        --------C----------------I--------S-----------P---
MON 21              C                  I   S                P
TUE 22             C                   S                   P
WED 23             C                S  I                P
THU 24              C         S        I            P
FRI 25               C     S           I       P
SAT 26                 *               I P
SUN 27        -------S---C----------P--I------------------------
MON 28              S       C  P       I
TUE 29             S      P     C      I
WED 30              S  P           C   I
THU 31              PS                 C
FRI 1 AUG 75       P   S               I   C
SAT 2              P       S           I      C
SUN 3         ------P--------S---------I----------C------------
MON 4                   P          S   I             C
TUE 5                        P         S               C
WED 6                             P    I   S             C
THU 7                                  P        S         C
FRI 8                                  I    P      S       C
SAT 9                                  I         P     S   C
SUN 10       --------------------------I--------------P--SC----
MON 11                                 I                CPS
TUE 12                                 I              C  *
WED 13                                 I           C     SP
THU 14                                 I        C        SP
FRI 15                                 I    C         *
SAT 16                                 I C         SP
SUN 17       ------------------------C-I-------PS-------------
MON 18                          C      I  PS
TUE 19                       C     P   S
WED 20                    C   P   S    I
THU 21                 C P   S         I
FRI 22               CP   S            I
SAT 23            *   S                I
SUN 24       -----*-S------------------I----------------------
MON 25            *S                   I
TUE 26            SCP                  I
WED 27             SC  P               I
THU 28              S C     P          I
FRI 29                S C        P     I
SAT 30                   SC            P
SUN 31       ----------------S-C-------I---P------------------

<*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*><*>
```

The OLD SOLDIER, now semi-retired, whiles away the hours crunching BASIC programs into his tiny PDP-8L. If you have a tiny EDUSYSTEM, send crunched programs to the Old Soldier, c/o PCC, P. O. Box 310, Menlo Park, Calif. 94025.

Build Your Own BASIC ~ Revived

WHAT IS TINY BASIC???

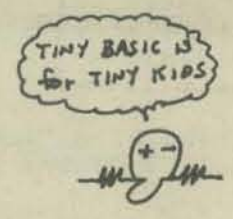
TINY BASIC is a very simplified form of BASIC which can be implemented easily on a microcomputer. Some of its features are:

Integer arithmetic 16 bits only

26 variables (A, B, . . ., Z)

Seven BASIC statements

INPUT	PRINT	LET	GOTO
IF	GOSUB	RETURN	



Strings only in PRINT statements

Only 256 line programs (if you've got that much memory)

Only a few functions including RND

It's not really BASIC but it looks and acts a lot like it. I'll be good to play with on your ALTAIR or whatever; better, you can change it to match your requirements and needs.

TINY BASIC LIVES!!!

We are working on a version of TINY BASIC to run on the INTEL 8080. It will be an interpretive system designed to be as conservative of memory as possible. The interpreter will be programmed in assembly language, but we'll try to provide adequate descriptions of our intent to allow the same system to be programmed for most any other machine. The next issue of PCC will devote a number of pages to this project.

* In the meantime, read one of these.

Compiler Construction For Digital Computers, David Gries, Wiley, 1971
493 pages, $14.95

Theory & Application of a Bottom-Up Syntax Directed Translator
Harvey Abramson, Academic Press, 1973, 160 pages, $11.00

Compiling Techniques, F.R.A. Hopgood, American Elsevier, 126 pages
$6.50

The TINY BASIC proposal for small home computers was of great interest to me. The lack of floating point arithmetic however, tends to limit its usefulness for my objectives.

As a matter of a suggestion, consideration should be given to the optional inclusion of floating point arithmetic, logarithm and trigonmetric calculation capability via a *scientific calculator chip interface.*†

The inclusion of such an option would tend to extend the interpreter to users who desire these complex calculation capabilities. A number of calculator chip proposals have been made, with the Suding unit being of the most interest.

Thank you for the note of 13 June, regarding my letter on the Tiny BASIC article (PCC Vol. 3 No. 4). It was with regret that I learned that the series was not continued in the next volume. Even though few responded to the article published, conceptually the knowledge and principles which would be disseminated regarding a limited lexicon, high level programming language are of importance to the *independent* avocational microcomputer community.

At this time, PCC may not have a wide distribution in the avocation microcomputer community. This could be possibly the cause for the low number of respondies. Never the less, this should not detract from the dissemination and importance of concepts and principles which are of significance.

The thrust of my letter of 15 April, 1975, was to suggest a mechanism for the inclusion of F.P. in a limited lexicon and memory consumptive BASIC. I hope that the implication that F.P. must be included was not read into my letter.

It is my interest that information, concepts and the principles of compiler/interpreter construction as it related to microcomputers be available to the limited budget avocational user. The MITS BASIC, which you brought up, appears from my viewpoint to be a *licensed*, blackbox program which is not currently available to: (a) 8008 users, (b) IMP-16 users, (c) independent 8080 users (except at a very large expense) or (d) MC6800 users who will shortly be on line.

Presently it appears that microcomputer compiler interpretor function langauges will be coming available from a number of sources (MITS, NITS, Processor Technology and etc.). However, few will probably deal in the conceptualizations which are the basis of the interpreter. Information which will fill the void in the interpreter construction knowledge held by the avocation builder, should be made available.

I strongly urge that the series started with Vol. 3 No. 4 article be continued. Possibly the hardware, peripheral, machine programming difficulties incurred by the microcomputer builder, is prohibiting a major contribution at this time. However, I would expect that by Autumn a number of builders should have their construction and peripheral difficulties far enough along to start thinking about higher level languages.

The previous objective for the article series sounds reasonable. It was not my purpose in submitting the letter to detract from the objective of a very limited lexicon BASIC, ie., to be attractive and usable by the young and beginner due to its simplicity.

If wives, children, neighbors or anyone who is not machine language or programming oriented is expected to use a home-base unit created under a restrained budget a high level language will be a necessity. It is with this foresight that I encourage the continuance of the "Build Your Own BASIC" series.

This issue aside, I would like to encourage the PCC to continue the quite creditable activities which have been its order of business with regard to avocational computing.

Michael Christoffer
4139 12th NE No. 400
Seattle, Wash. 98105

† For information on the calculator chip interface, write to: Dr. Robert Suding WOLMD, The Digital Group, P. O. Box 6528, Denver, Co. 80206.

Thanks for starting the series on building your own BASIC. This should prove very useful once the software limits have been established and the details start to come. There should be enough design information presented so that the techniques could be adapted to any computer.

Has anyone any recommendations for books or articles on designing assemblers, interpreters or compilers for minicomputers or microcomputers? I'm looking into the two mentioned on page 7 of the last issue. Are there any others?

R.E. Smallwood
20 – 12 St. N.W.
Calgary, Alberta, CANADA
T2N 1Y3

Just received the last issue of Vol. 3. What happened to "Write Your Own BASIC?" I sure was looking forward to it. I have –

Altair 8800 (256 bytes)
TVT I
Logiport I CRT terminal with modem.

I'm busy typing up a master for a 2K 2102-2RAM memory board for the Altair right now. Next is an I/O card.

I would think that learning to write an interpreter or translator would be invaluable experience to any computernik, whether his interest is software, hardware, or merely operating an existing system. Later, this year I hope to get the Altair programmed to do some amateur astronomy calculations and table look ups. It would be very handy to get real-time printouts of the positions of major celestial objects for any observer latitude, longitude etc.

Jonathan E. Tyler
5625 John R. Road
Troy, Michigan 48084

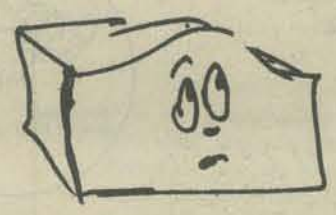
How's it going? I'm doing OK here, every Monday nite I get my mitts on a terminal, so I play around with games, algorithms, and such stuff. I'm getting into BASIC, and except for matrices, I think I know it pretty well. I really don' think I can use that new book of yours, since you chopped out the listings for some programs, and frankly the long ones are the ones where I need the game the most as I could write the smaller ones myself.

I liked the article on chips. Tiny BASIC looks to be pretty good. I might buy an Altiar 8800 if there is some simple way of getting a BASIC interpreter for it. That was pretty funny, the comment by my last letter in the paper, but not nearly as good as the excerpt about the Unknown Glitch.

Steve Follmer
623 Coram Rd.
Huntingdon Valley, Pa. 19006

The Mysterious and Unpredictable RND — Interlude

RANDOM NUMBER GENERATOR FOR THE INTEL MICROPROCESSORS



A Discussion by Gordon French

The Computer Hobbyist had Jim Parker's random number generator listing for the 8008. I entered it directly from the magazine, called it as a subroutine, and it coughed up all the random numbers that I cared to look at. The routine is very fast for the 8008, delivering about 300 random numbers per second. It is even faster for the 8080, though the version that I give here is only one byte shorter in length.

To use this routine, you call it as a subroutine. It returns the random number as a value in the range 000 to 255 (decimal) in the A register. This could be viewed as a value in the range +127 to −128 (decimal) if you care to look at it in that way. I have given it here in both codes, so that you can ke-switch it in in absolute if you haven't gotten your I/O boards from MITS yet.

There is one ugly about the routine, and that is that it will start to deliver numbers that are the result of having massaged the four bytes starting with location "shift". Nothing wrong with that, except that if you want the routine to begin with a different random number each time the program is loaded for execution, you must reach in and change one of the four bytes, or it will always generate the same random string for you. I'll leave you to figure out ways to make this routine generate random numbers randomly.

One way that I played with it was to set up a loop to call for a number, then check it to see if it was an ASCII code (i.e., 060 [octal] to 071 [octal]) and if it was, I printed it, counted it and inserted a carriage return and line feed if I had gotten to a count of 72 of the little devils.

I have located it absolutely at location 100 for my convenience. Those of you with a system that will allow you to enter it in mnemonics can relocate it where ever you like. In any event it will be easy to relocate since there are only a couple of references to hard addresses and the rest of it is all register to register.

One friend of mine commented that the 8080 version should be a lot shorter than the 8008 code, and wondered aloud if I had done something wrong. I told him that I wanted to exactly parrallel the two listings, but that in any even there would not be more than a couple of bytes difference because the two machines both handle their registers in very similar fashion, and differ more where jumps and calls are used. But in this case there is very little difference even if you shortened the 8080 code where you could. It is in any event, an object lesson for the brave souls who continue to work with (as I do) an 8008.

Those of you who plan to work up some computer games in assembly language may want to clip out this routine and save it until the need for a random number generator comes up.

JIM PARKER'S RANDOM NUMBER GENERATOR ROUTINE
FROM "THE COMPUTER HOBBYIEST" (VOL.1 NUMBER 5)

```
8080                                    8008

000100/ 041 LXI, SHIFT+3                000100/ 056  LHI 000
000101/ 145                             000101/ 000
000102/ 000                             000102/ 066  LLI 147 SHIFT+3
                                        000103/ 147
000103/ 006 MVI,B,3                     000104/ 016  LBI 010
000104/ 010                             000105/ 010
000105/ 176 MOV A,M                     000106/ 307  LAM
000106/ 007 RLC (TAG-RTOP)              000107/ 002  RLC
000107/ 007 RLC                         000110/ 002  RLC
000110/ 007 RLC                         000111/ 002  RLC
000111/ 256 XRA,M                       000112/ 257  XRM
000112/ 027 RAL                         000113/ 022  RAL
000113/ 027 RAL                         000114/ 022  RAL
000114/ 055 DCR,L                       000115/ 051  DCL
000115/ 055 DCR,L                       000116/ 051  DCL
000116/ 055 DCR,L                       000117/ 061  DCL
000117/ 176 MOV A,M                     000120/ 307  LAM
000120/ 027 RAL                         000121/ 022  RAL
000121/ 167 MOV M,A                     000122/ 370  LMA
000122/ 054 INR,L                       000123/ 060  INL
000123/ 176 MOV A,M                     000124/ 307  LAM
000124/ 027 RAL                         000125/ 022  RAL
000125/ 167 MOV M,A                     000126/ 370  LMA
000126/ 054 INR,L                       000127/ 060  INL
000127/ 176 MOV A,M                     000130/ 307  LAM
000130/ 027 RAL                         000131/ 022  RAL
000131/ 167 MOV M,A                     000132/ 370  LMA
000132/ 054 INR,L                       000133/ 060  INL
000133/ 176 MOV A,M                     000134/ 307  LAM
000134/ 027 RAL                         000135/ 022  RAL
000135/ 167 MOV M,A                     000136/ 370  LMA
000136/ 005 DCR,B                       000137/ 011  DCB
000137/ 302 JNZ,RTOP                    000140/ 110  JFZ 000107
000140/ 106                             000141/ 107
000141/ 000                             000142/ 000
000142/ 311 RET                         000143/ 007  RET
(TAG THE NEXT LOCATION "SHIFT")
000143/ 123 SET TO ANY VALUE            000144/ 123  SET TO ANY VALUE
000144/ 123 SET TO ANY VALUE            000145/ 123  SET TO ANY VALUE
000145/ 123 SET TO ANY VALUE            000146/ 123  SET TO ANY VALUE
000146/ 123 SET TO ANY VALUE            000147/ 123  SET TO ANY VALUE
```

I ENTERED THIS ROUTINE STRAIGHT FROM THE MAGAZINE AND IT WORKS BEAUTIFULLY. "TRY IT YOU'LL LIKE IT". FOR MORE INFORMATION, READ THE ARTICLE THAT THE COMPUTER HOBBYIST PUPLISHED. THEIR ADDRESS IS:

THE COMPUTER HOBBYST
BOX 295
CARY, NC. 27511

I'M HOPING THAT THE GUYS IN CARY COME UP WITH A LOT MORE OF THIS KIND OF THING. I USED IT AND ENJOYED DINKING WITH IT. IF YOU HAVEN'T READ THEIR SHEETS, BETTER GET IN LINE.

MANY THANKS TO THEM FOR LETTING BOB ALBRECHT PURLISH THIS PLAGERIZED VERSION OF A GOOD PIECE OF WORK. ALSO MANY THANKS TO ED HALL WHO CHECKED OUT THE 8080 CODE FOR ME.

SOLDERING?

It's not the only way.

BY ROBERT MULLEN

Hand soldering is a quick, easy and effective method for connecting conductors. A well made solder connection is neat, strong and has a very low electrical resistance. P C boards and most kits require soldering. Machine soldering of P C boards is very popular for large volume production, but soldering makes circuit changes and re-use of the components difficult. Heat from soldering can damage components and construction is slow.

The main alternatives to soldering are socket strips and wire wrap. Socket strips are plastic boards with small sockets on a 1/10 inch grid. The sockets are bussed together to allow the rapid connection of DIP and other standard components. Jumpers are made from stripped solid hook up wire, usually no. 22 AWG. The wire is pushed into a socket to make the connections.

Socket strips are made in sizes which range from one which holds a single 14 pin DIP to large assemblies capable of holding several hundres components and furnishing regulated power, signal generation, switches, indicators, etc. They are generally used for circuit development as they are not easily adapted to building multiple copies or connectors to other equipment.

They are initially expensive but allow repeated use of and do not damage the components; you can even save and re-use the wire jumpers. This makes socket strips ideal for educational use where many circuits are to be studied.

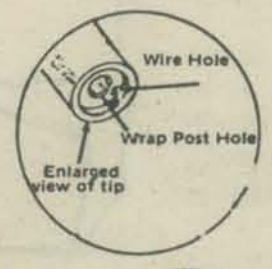
A couple of places to write for catalogs:

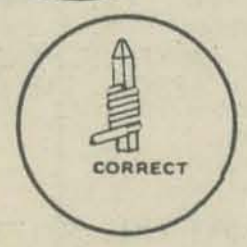
Continental Specialties Corp.
Box 7809
San Francisco, Ca. 94119

A.P. Products Inc.
Box 110
Painesville, Ohio 44077

Wire wrap was developed in the 1950's by Bell Labs as an alternative to soldering. Wire-wrapping consists basically of winding a number of turns of wire around a metal post with at least two sharp edges. In practice, the metal post has evolved into a standard 0.025 inch square pin. With the correct wire and tension during wrapping, a clean metal-to-metal contact results. The corrosion resistance, mechanical stability and conductivity are good enough for the technique to be used in military equipment.

Wire-wrap is widely used in industry for proto-type work and, using semiautomatic and automatic machines, for short run production. Wide usage has brought with it a broad range of hardware such as tools, DIP sockets, edge connectors, and even whole logic boards.





A wrapping tool is a pencil sized shaft with two holes in the end. The larger hole fits over the wrap post; the smaller hole fits over the wire. Wire sizes are 26, 28, and 30 guage. The tool can be turned by hand, or there are a variety of power drives available. For production work electric and pneumatic tool drivers are common. In proto-type work, battery powered drivers avoid the inconvenience of a trailing cord.

To make a wrapped connection the wire is stripped back far enough to give at least 6 complete turns, the end of the wire is put in the wire hole of the tool up to the insulation, and the tool is slid over the post and rotated. The first couple of turns are made with no downward pressure and with the free end of the wire held securely. When the wire catches on the post, pressure is applied. The correct pressure will result in a wrap with the wire tightly wrapped about the post. Too much pressure and the wire will wrap over itself, too little and the wire will spiral up the post. It takes a little practice to develop the feel to produce the correct wrap.

EXAMPLES OF INADEQUATE WRAPPED POSTS

Wire-wrap is best applied to making equipment to be used. The variety of hardware available makes connecting to existing equipment easy. Wire-wrap can be converted to hybrid P C boards, with power, ground, connectors and any stable circuitry on copper and circuits still under development in sockets and wire-wrap.

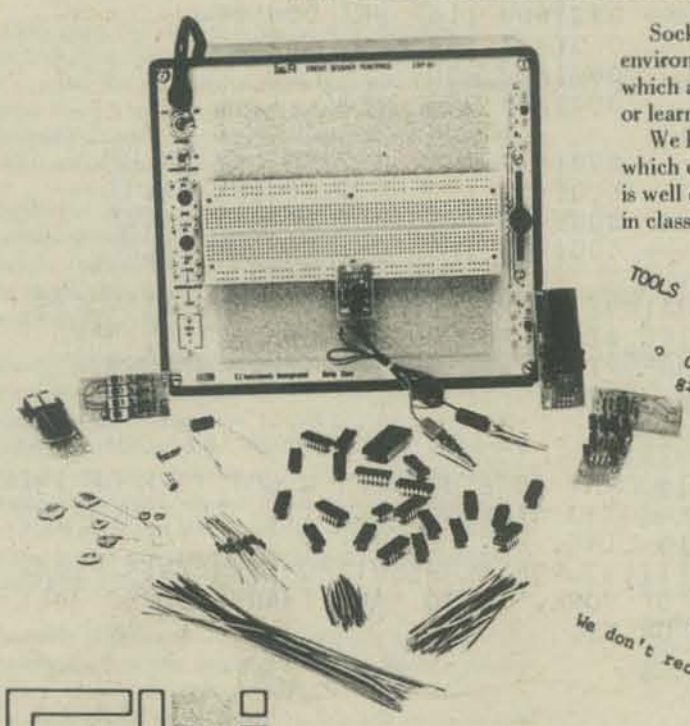
Wire-wrap tools have been unbelievably expensive, but an excellent battery powered wrapping tool at a hobbyist price, can now be had from:

$41.95

Godbout Electronics
Box 2355
Oakland Airport, Ca. 94614

Another supplier of wire-wrap tools:

O K Machine and Tool Corp.
3455 Conner St.
Bronx, N.Y. 10475



Socket strips allow assembly and disassembly of circuits -- exactly what is needed in a teaching environment. E&L Instruments have used this ability as the basis for a teaching system -- and one which according to a very reliable source, is probably the best package on the market for teaching or learning digital logic.

We have just received a set and can see what he means. Not only is there a wealth of experiments which can be performed, but the profusely illustrated text which came with it (almost 1000 pgs.) is well enough written to be even useable as a self teaching course. Bob Albrecht will be using this in classrooms this fall. Expect an article on his experience around December.

TOOLS

the RUGBOOK T

° Only three tools are really necessary for all of the experir given in this laboratory workbook,

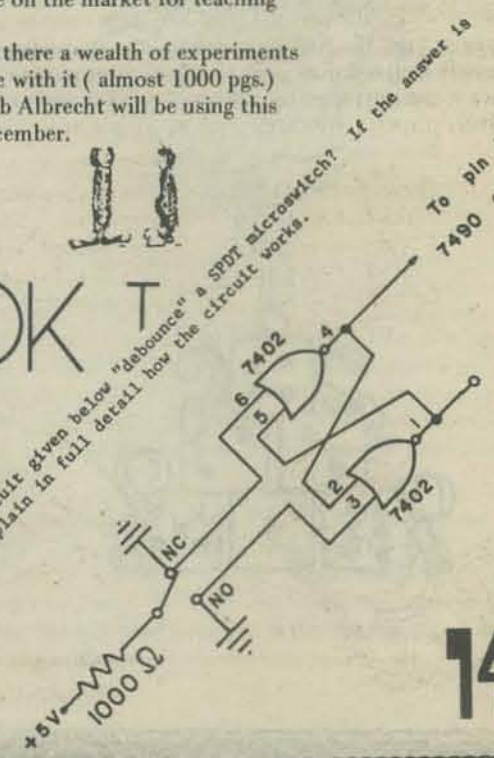
a pair of pliers [We currently prefer the XCELI chain nose plier with spring, No. 72CG.]

a screwdriver [Any inexpensive screwdriver the 2" approximately 4" to 5" long and has a tip width 1/8" or a little less is quite satisfactory.]

a wire stripper/cutter [We prefer the XCELITE wire stripper/cutter, No. 101-S.]

We don't recommend that any other tools be purchased.

Will the circuit given below "debounce" a SPDT microswitch? If the answer is yes, please explain in full detail how the circuit works.



Think of a Number

by SIVASAILAM THIAGARAJAN

NUMBER OF PLAYERS: Three or more. Five or six make the best game.

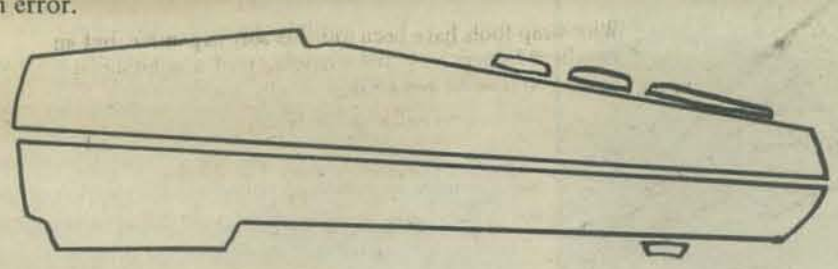
APPROXIMATE TIME REQUIREMENT: Depending upon the endurance of the players, one to five minutes. Game is replayed with different players assuming the role of the storyteller.

SKILLS INVOLVED: Basic arithmetic operations, solving equations involving a single unknown quantity, and making inferences.

CHANCE FACTOR: None.

PRELIMINARIES: Before the play of the game, one player is chosen to be the storyteller. Since a complete round of games involves every player in this role, it does not matter how the first storyteller is selected. During subsequent rounds of the game, players take turns assuming this role.

In spite of its apparent simplicity, this game lends itself to all sorts of ingenious strategies. Each round of the game has a player in the role of the storyteller who selects a single-digit mystery number. He keeps manipulating this number by adding, subtracting, multiplying and dividing it by other numbers. He tells the others what number he is using and what he is doing with it. But he never reveals the results. Other players attempt to keep pace with him until their information-processing mechanisms can hold no more. They individually ask the storyteller for a peek at the display and try to trace back the original mystery number. If successful, the player's score is the number of rounds he has endured. The object of the game is to outlast the other players without making an error.



PLAY OF THE GAME:

1. The storyteller selects a single-digit mystery number and punches it into the calculator. She also locks it up in the memory of the calculator if there is one. If not, she writes it down secretly on a piece of paper.

Charlotte selects 7 as her mystery number.

2. The storyteller now performs a series of steps with this mystery number. During each step she may add, subtract, multiply or divide using any single-digit number. She has to make sure that the results of these operations are never negative, fractional, or greater than 50. Nobody sees what the storyteller does on the calculator or the partial results on the display. However, she informs the others exactly what number she is using and what she does with it. She also keeps count of the number of steps taken. Other players try to keep track of what is happening to the original number.

These are Charlotte's statements as she manipulates the mystery number 7. "Step 1. I am adding 9 to the number." She gets 16 on the display. "Step 2. I am subtracting 2." This is a wasted move. She gets 14 on display. "Step 3. I am dividing by 7." This is a strategic blunder as you may have figured out already. The display now read 2. "Step 4. I am multiplying by 3." The display now reads 6.

3. Each player keeps track of the various changes in the mystery number as long as he can. He may stop the storyteller anytime he is ready to quit. The storyteller shows him, and only him, the number on display. The quitting player now has to retrace the mystery number and secretly reveal it to the storyteller either by writing her a confidential note or by whispering into her ear. If the mystery number is correctly guessed, the player gets a score equal to the number of steps taken so far. If incorrect, he scores a zero.

As Charlotte works through her story, here's how different players keep track of her moves: Thiagi, who is not very mathematical and has not played this game before, keeps mumbling to himself, "Add 9, subtract 2, divide 7, multiply 3 . . . " Harold, who has taught math for three years uses an algebraic approach and stored the information thus: "X . . . X plus 9 . . . X plus 9 minus 2. That's X plus 7 . . . X plus 7 divided by 7. The whole thing's divided by 7 . . . The whole thing's multiplied by 3 now. That makes it 3/7 times X plus 7 . . . " Lucy, who loves numbers, starts out the same way Harold did. However, in step 3 when Charlotte divides by 7, her analytical brain functions this way: "The mystery number is less than 10. It has 7 added to it and is divided by 7. It still remains a whole number. So it could only be 7 in the first place!" Since she has discovered the myster number itself, she doesn't have to listen to Charlotte's story any more.

After step 4, Thiagi could not hold his chant any longer and he says, "I'm ready to quit." Charlotte shows him the 6 on display. Thiagi now does all the operations according to his chant, but in the reverse order and in the inverse fashion. In other words, he divides 6 by 3, multiplies the result by 7, adds 2, and subtracts to get the original mystery number of 7. This is hard work and he could have goofed almost anywhere. However, his luck holds out and he whispers the correct answer to Charlotte. She confirms the answer and awards him a score of four points.

Charlotte continues her story for the benefit of the other two players: "Step 5. I am adding 3." Harold loses his concentration now. The last thing he remembers is 3/7 times X plus 7. He now thinks, "3/7 times X plus 7 plus 3. That makes it 3/7 times X plus 10. As you can see he makes an error here: The equation should have been (3/7(X + 7)) + 3; Harold decides to quite and asks Charlotte for the display. He sees the 9 and tries to get back to the mystery number by solving his equation for X. He does this correctly but since his equation was incorrect gets a funny result of −2.33. Charlotte gives him a score of zero.

4. Anytime during the game a player may say "I want to go blind" and reveal the mystery number without looking at the display. This gives him an automatic score of 20 points.

Lucy could have done this before but she did not want to alert the others to the fact that there is enough information to infer the original number. Now that everyone else has dropped out of the game, she triumphantly reveals 7 as the mystery number. Since she is absolutely correct, she scores 20 points immediately.

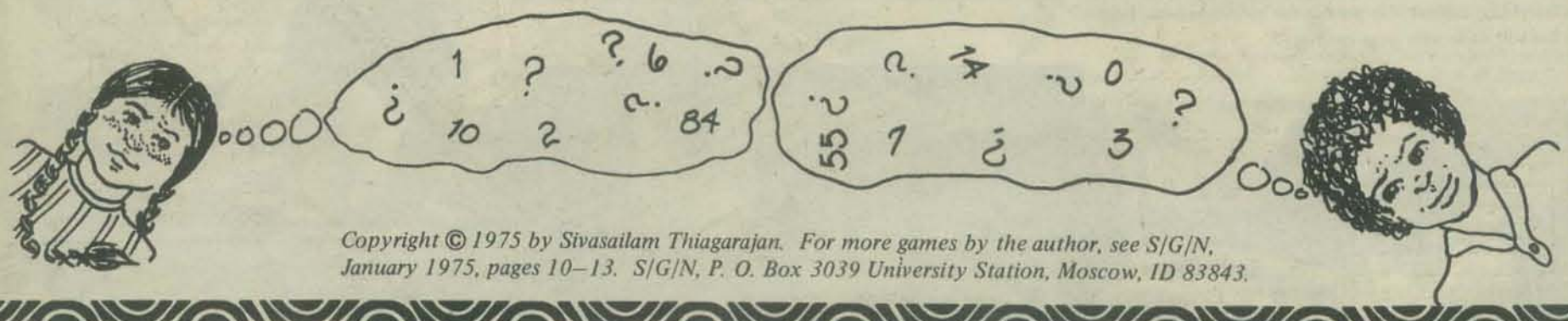
The game is repeated with Thiagi assuming the role of the new storyteller.

VARIATIONS:

1. During the first two or three times the game is being played, permit players to use paper and pencil. It still permits all sorts of strategy.

2. To compensate for discrepancies in mathematical competencies, players may set up minimum number of steps for each. Thus among the players in our sample game, Lucy may have a minimum of 5, Harold a minimum of 2, and Thiagi no minimum. A player's score will be the number of steps beyond this minimum.

3. With children, the game may be limited to addition and subtraction.



COMMUNITY MEMORY

KEN COLSTAD &

ABSTRACT

For the last year the Community Memory Project has been demonstrating the potential of computer-based *public access* communications media with a small network of public terminals in the San Francisco Bay Area. From any terminal it was possible to search a common data base using boolean combinations of keywords or to add and index new information/messages of whatever nature the user desired. Both the ease with which the public accepted the service and the imaginative uses to which it was put were surprising and gratifying. The project is currently developing hardware and software systems to move the idea from an externally financed experiment to a cheap, self-sufficient service available in all the neighborhoods and to all the cultures of the Bay Area. These systems would supply the basic tools for establishing similar services elsewhere, and provision is being made so these regional networks could be linked to form a continental information sharing network. It is hoped that the project will serve as an inspiration for using the computer technology to meet real human needs rather than to make money.

After twenty-five years of computer development, the question is still open as to whether this technology can be directly useful to the public. People at present generally believe that computer systems are used on them rather than for them. Could computer information systems be accepted and used by the public? In most information-handling systems, people have no control over the way data about them are acquired and used. Information in these systems is used for monitoring of people by institutions, and is often regarded as useful if it is negative.

The few public-access systems are vertically organized, conceived primarily for delivery of computer-aided instruction and other pre-selected information, as thoroughly editted as in other forms of mass communication. The possibilities of horizontal, person-to-person data acquisition and delivery have not been explored.

Such a horizontal system would allow the public to take advantage of the huge and largely untapped reservoir of skills and resources that resides with the people. One-to-one communications media such as telephones and letters create no new links, while one-to-many connections such as television, newspapers and bureaucracies inevitably restrict the flow of information through their offices. Since political and economic power follows the lines of communication, the potential for abuse is tremendous. A large pool of information, freely accessible and amendable through public terminals, is one of the few systems proposed for many-to-many communications.

A critical context for use of such a system would be in community based information centers rather than terminals located only in private homes. This might counteract the tendencies toward fragmentation and isolation so visible in today's society by significantly augmenting environments where small groups of people congregate and interact on an informal basis.

For the past year the Community Memory project has been demonstrating the potential of computer-based public access communications media with a small pilot network in the San Francisco Bay Area. From three publically located terminals it was possible to search a common data base for information or to freely enter new information or messages. The public accepted the service with remarkably little hesitation and put it to a much broader range of uses than was anticipated, proving that given the tools, the public will not only provide for its own information needs but will do so with great creativity.

This was a crucial question for the organization which spawned the system. Resource One, Inc. of San Francisco is one of the few public service computer centers in the country, a non-profit corporation devoted to charitable and educational uses of data-processing technology. Resource One had available an XDS-940 timesharing computer and ROGIRS, an efficient keyword based text retrieval package based on the MIRS system developed by Robert Shapiro of META. The software was modified to simplify the command structure for public use and to improve the security of the data and of other system users.

To use Community Memory, the user would type the command ADD, followed by the text of the item, and then by any keywords under which he desired the item to be indexed. To search for an item, the user would type the command FIND followed by a logical structure of keywords connected with AND's, OR's and NOT's.

The first port to this system was installed without fanfare adjacent to a bulletin board in a non-profit community record and music store in Berkeley. People were delighted by the chance to put a computer to use, frequently commenting that "it's about time!" They encouraged their friends to use the system, instructed one another in its use, and seemed fascinated as much by the possibilities of the medium as by the technology itself. This level of acceptance was not confined to the relatively sophisticated student area, but carried over to later installations such as one at a library in San Francisco's polyglot Mission District.

Initially the location of the terminal and its popular characterization as an 'electronic bulletin board' determined the public's expectations and uses of the system. Installed during the August housing crunch, it became immediately useful in the students' searches, with the rate of success growing with the size of the data base. Musicians, always in search of others with whom to practice, entered themselves and their special areas of interest. Instruments were bought and sold, producers found new opportunities, and groups advertised their availability. New groups, in fact, were often assembled on the spot from leads found in the data base, and from people waiting around for their turn to use the terminal. Similarly, people used it to assemble car pools, organize study groups, find chess partners, and pass tips on good restaurants. Interesting and unanticipated uses developed: poems, graphics, dialogues among strangers, and items most analogous to letters to the editor, but much freer in content and form: instant publication by a 'very small press' had become available to all who claimed literacy.

The rate of use of the system was fairly high and constant in relation to the environment of the terminals. About fifty searches and ten additions occurred each day at each location. Given the length of individual sessions with the system, this was at least one-third the maximum capacity of a terminal.

The crucial factor in determining the manner in which the system was being used was the rate of success, which in turn was determined by the data density for each subject area. A bootstrapping effect brought the density up slowly to a critical level, after which usage rose rapidly to a maximum level for that application. This critical level was never reached for certain roles in which the system would be uniquely valuable, such as a skills bank, learning exchange, forum for ad hoc organization, or barter marketplace. Since no institutions have filled these information needs, they are not generally expected to be met, but a significant number of users independently innovated these applications of the system.

The bootstrapping principle was self-evident in operation, and a number of individuals stimulated the process with bulk entries in their own special interest areas. These gratuitous offerings of information contributed strongly to the richness, diversity, and utility of the data base. Information degrades, however, and the responsibility felt by these users for maintenance and updating could not be effectively dealt with or assessed. To safeguard against unilateral censorship or destructiveness, the public had no editing privileges, although a number of people clearly could have been trusted to shepherd parts of the data collection. The system provided for maintenance by requesting a deletion date at the time of the addition, but this proved inadequate for all but the most 'classified ad' type of entry.

Malicious and obscene items, trivia, and misinformation represent the major opportunities for abuse of the system. In practice this kind of misuse was not prevalent, but scanning for it increased the maintenance responsibilities of the pilot project staff. An attitude of 'caveat emptor' has been advocated in this regard, since the content and relevance of the items the user finds can never be guaranteed. The editorial processes that have evolved in other media are not completely successful in this context, nor are they readily transferable.

& EFREM LIPKIN

a public information network

lectrical and Electronics Engineers, Inc.
OMPCON 75.

Loving Grace Cybernetics
1609 Virginia St.
Berkeley CA 94703

Other inherent problems appeared due to inexperience on the part of the users with typewriter keyboards, spelling errors, and misunderstanding of the keyword concept. The social interactions around the terminals have been the only way of dealing with these difficulties.

The other deficiencies encountered in the operation of the pilot system can be effectively dealt with through redesigned software. The primary consideration in current design plans, however, is maintenance of conviviality in the interactions with the users. People must gain a sense of understanding of and control over the system as a tool. While it must command sufficient intelligence to recognize and respond to the most naive user, that intelligence should be directed toward instructing him, demystifying and exposing its own nature, and ultimately giving him active control. Meeting this criterion without placing excessive demands on the user deeply tests the system designer's ingenuity.

This is especially relevant in the case of the current design strategy, which includes the implementation of a tree structure of categories as a parallel and alternative mode of searching for items. This would allow users unfamiliar with the system to browse through a structured environment of hierarchically categorized items while enabling more experienced users to search directly on content with the system in a more passive mode. Any such categorization scheme is necessarily biased by the paradigm with which the designer interprets and organizes the world. Minimizing this effect complicates the system and challenges the design group.

Other innovations under development include the implementation of named fields to aid narrowing the searches by date and value. Item ownership will allow 'information shepherds' and organizations such as switchboards and other referral agencies to maintain subsections of the data base for their own use while sharing it with the public. Dialoguing and conferencing will be more explicitly supported, while games and other special purpose programs will be available to various users.

The pilot system, supporting few terminals on a large, expensive general-purpose time-sharing computer, was not economically reasonable. Through careful mathematical analysis it has been determined that by using an optimized file structure, good searching procedures, and a thoughtfully coded mostly core resident program, more than 64 simultaneous users could be serviced by a 24K mini computer the speed of a NOVA or PDP 11/40. Such software is currently being developed along with custom terminal multiplexing hardware which will greatly reduce the load this many terminals place on the CPU. With the broad base for capital and maintenance costs this system provides and the use of the low-cost, people-oriented Tom Swift Terminal described elsewhere at this conference, costs should be less than $2000 per public access site.

Each of these minicomputer systems will be capable of networking with others, exchanging information of a non-localized nature, and providing a nationwide conferencing medium. Groups such as Infact in Vancouver, B.C. and the Boston Children's Museum are contributing to the design of the mini-system while testing concepts with their own systems based on Community Memory. A cooperative effort seems the correct way to bring about systems for information sharing.

The cooperative use of technology to meet human needs, rather than its competitive use to create lucrative mass markets in electronic elaborations of simple devices and services, is the basic goal of the Community Memory project. This sort of direction is a sadly rare style among engineers, programmers, analysts, and the people who coordinate their work. But the issues of how and for whom the technology will be made to perform are becoming ever more critical. They play a deep role in the continuing economic, ecological, political and energy crises. These issues must be dealt with by both the people who have mastered and currently control the technology and those people it is claimed the technology is serving. But the heaviest responsibility lies with us, who create with the technology, to be conscious of the significance of our creations and to actively make sure that they are directed toward the greatest good.

REFERENCES

1. Christopher Beaty: *The Journal of the Bay Tea Company (The Public Memory Papers)*, 1974, privately published and distributed.
2. Ivan Illich: *Deschooling Society*, Harper and Row, New York, 1971.
3. Ivan Illich: *Tools for Conviviality*, Harper and Row, New York, 1973.
4. *Mediation*, vol. 1, no. 1 , Loving Grace Cybernetics, Berkeley, CA.
5. Laurence I. Press: "Arguments for a Moratorium on the Construction of a Community Information Utility", *Communications of the ACM*, vol. 17, no. 12 (Dec 1974).
6. Robert M. Shapiro: *A Handbook on File Structuring*, RADC-TR-69-313, Applied Data Research, Inc., New York. Sept 1969.
7. Kenneth J. Stetten and John Volk: *Interactive Television*, M72-200, The Mitre Corporation, Washington, 1973.

This article and several others are included in the Journal of Community Communications, Vol. 1, Issue 0, published by LGC Engineering, 1807 Delaware St., Berkeley CA, 94703.

The purpose of the Journal is "to encourage and develop the dialogues which are starting concerning the desireability and possible forms of low - or non - hierarchical communications systems which can be created, shaped, and used by people in their daily lives as members of communities".

A free copy of Issue 0 is available from LGC if 20 cents postage is included.

Kit$ and Hardware

WHAT IS A KIT?

A kit for an electronic device usually consists of one or more printed circuit boards, a bag of bits and several sheets of paper with instructions. Often a case for the device is included, or the materials needed to make one. Solder is sometimes included too.

Most can be put together with a soldering iron, a pair of wire cutters, a pair of pliers and a screw driver. Homes are presumed to have these, so they aren't included. Assembled, a small kit will usually perform (with at least a family resemblance to the item you thought you were getting when you read the advertisement) trouble free from the first time you apply power. Any necessary adjustments for tuning will normally be no more difficult than the assembly and require no special apparatus.

Advanced kits are more complex, but generally they take longer to put together rather than require more skill. Tuning can become a problem though, since advanced kits often require the use of electronic instruments. If you can't borrow them, your only recourse is your local TV repair shop. However, the cost isn't much considering that your kit probably cost several hundred dollars. The real catch with a large kit is that it is much less likely to work first time – and this isn't necessarily either your fault or that of the manufacturer. Those of us who are not used to electronics components expect items to be as they are described. We expect a 5/8" nut to fit a 5/8" washer. Semiconductors are different. The manufacturer produces a specification sheet which is replete with digrams showing nice clean square pulses ⎍⎍⎍. What you get is actually more like ⎍⎍⎍. And you don't even get that if your component is one of the percentage that doesn't work at all. Don't heap blame on the maker – if he produces a high quality item, it merely means that almost all meet the specs. Testing is costly; the user pays for it, and it can't detect every faulty item anyway. Typically there is a high failure rate for the first few hours of use, dropping to a very low one and then eventually rising again as the product approches the end of its life span. This aging takes place in your machine because you can't afford to pay the manufacturer to age it for you

So what do you do? If it doesn't work you fix it. Whether you can depends on what it is - a loose wire, solder bridge or obvious component failure yes, but often you can't. Fixes require understanding what is wrong, and this the kit doesn't give you. Fixes usually require equipment which you probably don't have. A simple device you may well fix, but increasingly, kits are anything but simple devices. Integrated circuits enable a child to assemble a device which may be electrically more complex than a WWII radar installation.

Can you build a kit? Sure despite the size and complexity of even the largest kits, they all go together piece by piece; one easy step after another. Start with a little one though, the big ones require many hours of work, and it's easy to become discouraged, since nothing will work until it is all finished

There are lots of reasons advanced to justify the purchase of a kit. Some are valid. Some are rationalizations. Some are plain fallacious. Mostly kits are bought because some phrase in the advertising clicked with someone's dream. This process of selection defies logic, but knowledge of good and bad designs makes it more likely that an unconscious choice will be a good one.

Many kits are purchased because the buyer can't afford a professional unit. The impression is often fostered that the kit is cheaper because the manufacturer is saved the cost of assembly. But when it is assembled IT ISN'T ALWAYS A PROFFESSIONAL UNIT that THE BUYER GETS. One of the sad truths of the world – almost everything that is manufactured is cheaper and better than the nearest equivalent that the amateur can build. If you buy a board from DEC for your PDP8, you will find that it's expensive. If you build your own, you will save – but in a little while the contacts on your board will oxidise and corrode causing trouble. While the gold plated ones from DEC are still OK. Some of your components may give trouble. So may some of the ones DEC uses – but they got rid of any which were not quite up to specs, sold them to the guy who sold them to you.... And you don't have the equipment to test them with anyway.

If you have a problem, you can call a major manufacturer and get help. Part of the higher cost of their boards goes to pay the salary of the guy who answers the phone. If you paid a small kit manufacturer the same money, he could afford to do the same but you didn't, so don't expect the same level of support.

This is not always true, see our report on Processor Technology, but take note of this quote from the Micro 8 Newsletter:

> *" Let us repeat! When you send off a check to a supplier, kiss it goodby, because you may never see that money again. What's almost as bad is if the check gets cashed and you wait months for delivery on the items you need now. If a guy's advertising looks too good to be true, it probably is. If it looks like he is offering too many hard to get items, beware. If he can supply them, other people could also, and they wouldn't be hard to get. If the prices are much lower than other outfits, he's probably selling junk and its hard enough to get these complicated computer systems running without having to find defective parts as well. What do you do when a guy offers something you have to have and he is the only one that offers it? I don't know. But you'd better clarify everything by telephone and or letter before you send your money.*
>
> *If they accept Mastercharge of BankAmericards, you've got several things in your favor. At least some bank recognizes them and part of the agreement they sign is that they will submit information on when and how items were shipped. If you don't get them, you just stop payment on the bill at your bank."*

There is one final criticism of kits: If a kit is not part of an instructional course, there is usually very little to be learned about electronics by building one. The instructions may read, "Orient the diode as shown in DIAG.4, solder." This doesn't tell you its purpose in the circuit, much about it as a component or even why the orientation matters.

HOW MANY BITS?

Bitter arguments rage on this topic. For the hobbyist the choice is pretty much 8, 12, or 16, and each has its disadvantages. The best choice will be a compromise and will depend on the use to which you put your computer: math, character oriented or machine control. The size of a number defines the use to which it can be put, and the smallest useful number is six bits or 64. This gives a code sufficient to express numbers, a few special characters, an upper case (all capitals) alphabet and twenty control characters. This is enough to drive a TTY but inadequate for other duty and inherently produces text which is unpleasant to read. It has one other thing going for it - groups of three digits suit expression in octal, which is much nicer to use than hexadecimal, which fit groups of four. For reasons which will be apparent later, it is best to combine two 6 bit bytes to produce a machine which uses 12 bit words e.g. the PDP8's.

Seven bits is awkward to use, being odd, so the next standard is 8 bits, or 256. This does provide a code for any pattern we might want to print, plus all the control codes we might read. If you want to control a machine, it will provide steps of ¼%, which is fairly smooth, but doesn't match the 10 bit output from most analog to digital conversion. Its real disadvantage lies in the fact that 256 is too small to be directly useful for math.

Experience shows that 6 decimal digits are enough precision for general purposes, which is why BASIC is standardized at that. But this represents 24 bits - so with an 8 bit machine the processor must lose time addressing memory, particularly since it must specify a memory address each time, normally two more 8 bit bytes. The processor, to get one 24 bit value, puts two bytes on the address bus, gets one back on the data bus, puts two out, gets another back, puts two more, gets the last one back. This is tedious - but it is nothing to the number of steps necessary to even just add two numbers, since the processor then has 6 bytes to work with, plus any carries that might be generated.

Obviously the 12 bit machine will only have to go to memory twice to get 24 bits, and it will be simpler to perform mathematical operations. For machine control, 12 bits is a fair match to the 10 bits from an A to D converter and gives the precision of better than 1 in 1000, which is fine. Each 12 bit word can also represent two 6 bit characters, so the machine is well suited to running in BASIC on a TTY. But it can't easily talk in lower case (small letters), and if you want to do business math, you can't easily express $10,000.00 because you have only 6 decimal digits in 24 bits. And when the processor tries to talk to memory it has real trouble. 12 bits is about 4000 decimal, so that is the limit of the size of memory which can be directly addressed - and indirect addressing is an unbelievable pain.

The 16 bit machine can directly talk to 64,000 memory locations, enough for most purposes, and does so in a simple operation since it has 16 bit words which suit 16 bit addresses. Two such words provide 32 bits of precision, or about 9 decimal digits (fine for business math). One 16 bit word holds two 8 bit bytes, so it can be use the desireable 8 bit character set. The processor has lots of instructions, and is, therefore, easy and flexible to program. So, this is the way to go – right? Well not necessarily, because at somethings a 16 bit machine is big enough to begin to be inefficient and it costs quite a lot more.

If you want to work with single characters, your 8 bit machine will run only slightly slower and use half as much memory. Your 16 bitter has to remember not only the location of the word, but also which byte, a programming complication, and it isn't as efficient in the use of the memory it stores instructions in. Both use 2 bytes to store an address, but the 8 bit machine uses only one for an instruction while the bigger machine has to blow a whole word. On the other hand, the 8 bitter may need three or four instructions to do something which the larger machine does with one. Similarly, with math operations. If you need 32 bit precision, the 16 bit machine is tops. But if you don't, and you don't to run BASIC - the bigger machine may be of little advantage. It does better math than the 12 bit machine, because of the greater precision, but it is not necessarily faster. For some operations, the speed is the same - it takes a fat man no longer to go through a fat door than it takes a thin man to go through a thin door - but for other operations it depends on the number of bits involved, 32 verus 24. And the cost and complication increase sharply.

I/O devices are arranged to handle single characters, usually 8 bit. This involves expense and complications since the 16 bit processor talks in two character words and something has to translate. The processor itself is more costly, and within the machine there are typically twice as many components to route the wider signal.

So it comes down to the following choices:

With an 8 bit machine, you can run BASIC, machines and handle text quite well and cheaply. But for business use, you must extend your precision and for involved math, you must also have patience - it will take time.

With a 12 bit machine, you can efficiently run BASIC and machines, have real problems addressing memory, a lousy character set, a slight increase in speed over 8 bit, and all that PDP8 software.

With a 16 bit machine, you can do serious math work at reasonable speed, but for simple math or character manipulation there may be little advantage to balance against a sharp increase in cost and complexity.

In the 8 bit hobby field, , these are, at present, three LSI chip contenders, the Intel 8008, the Intel 8080 and the Motorola 6800. The 8008 is the oldest and there is quite a bit of software for it. The instruction set is small; a disadvantage, but its real problem is lack of speed. With the cost of the faster 8080 dropping, rapidly, there seem little reason to buy the 8008 in preference. But it remains a competent processor, and there will soon be a lot of memory chips that are very cheap becausethey are too slow for the faster CPU's, so if the price is right....

That leaves the 8080 and the 6800. There is little too chose in speed between them. The 8080 lends itself to much better memory bus and I/O arrangements. It has one accumulator plus a selection of registers while the 6800 has two accumulators, a clear plus for many arithemetic operations. The 6800 also has elegant interrupt handling features built into the chip - to match them the 8080 needs a seperate board. The Motorola product range includes a number of particularly attractive chips, but the Intel range is much more extensive and software for the 8080 has a long development lead.

8080 NEWS: MITS is still swamped with orders, having problems with suppliers, run ragged with phone calls and has shipped some copies of 8K BASIC. They are being second sourced widely - see our report on Processor Technology.

8008 NEWS: Prices are falling.....falling.....

6800 NEWS: The race is on for the first kit based on this chip - odds on favorite is SPHERE - see report. We know of others and will publish a list and progress report next issue.

12 BIT NEWS: Contenders here are the PDP 8A or the new chip from INTERSIL. Currently no kits, and either needs an OEM buy. Write to C. Richard Corner, 514 So. 9th St., Moorhead, Mn 56560 of the Micro 8 group who is trying to put one together. (Also for LSI-11)

16 BIT NEWS: LSI-11 boards from DEC - but OEM only, see note above. Two new kits from Bill Godbout, see report, one a near NOVA, the other an MSI-11. Nothing else in sight but these look so good we are dropping our interest in an OEM buy on the LSI-11.

TV Typewriters

TV typewriters are not very complicated and should be very easy to make from kits or plans, but they seem perculiarly susceptible to trouble. We have yet to hear of anyone one making one and having it work first time. Mostly this has been owing to broken solder traces, solder bridges, or faulty components - one or two in most kits.

Surprisingly, this doesn't seem to bother people. No one was sharply critical of Southwest Technical, and an overwhelming majority of owners of their kits were satisified that they had received value for their money. This may reflect the comparative sophistication of Bay Area electronics hobbyists. Few here are surprised when components don't work — even reputed "tested" ones straight off the shelf. And enough is heard of things that go on in the electronics industry to develop both a background against which the honesty and technical competence of a company can be judged, and a sympathetic understanding of the problems of the manufacturer.

The location of faulty IC's generally proved fairly straightforward. Though the friend with a scope came in handy, most owners felt their trouble shooting could have been done with a good multimeter. Failures are sometimes mysterious. One father/son combination, building a SWTP CT1024 powered it up and found it had a couple of faults. Rather than struggle, they promptly bought a Heathkit scope kit, built it and found that unserviceable too. After three faults had been fixed on that, they got back to the TVT and located a faulty IC, which they replaced with a new, high quality one which they had tested to make absolutely sure that it was within specs. Didn't work in the circut so they replaced it with a nasty cheap surplus one, sans pedigree, which worked fine. One solder bridge later their TVT was up, running and, according to their shiny new scope, propagating the right pulses to the right places. Owners delighted and happy with both kits.

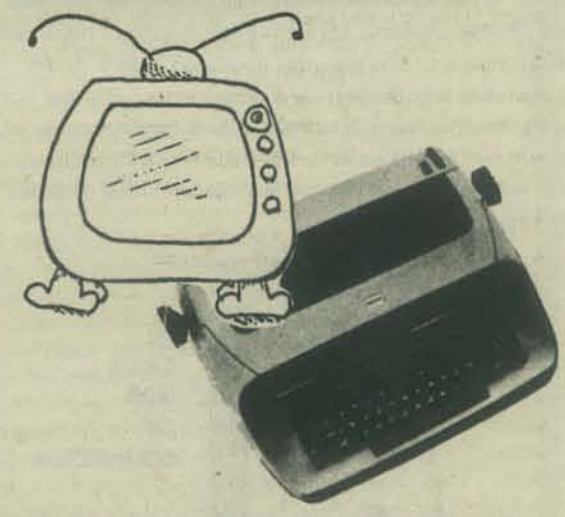
The only kit manufacturer with much track record for TVT's is SWTP. The newer kits are better in all respects than their first ones, and owners seem satisfied with them. They have very low density boards, so are easy to work on, but the assembly instructions are primitive, particularly on the subject of actually connecting the thing to a TV. Two design weaknesses stand out - oxidizable copper contacts on the keyboard and the use of Molex connectors. These latter are pins soldered to the PC board on which other boards are mounted. Thus the smaller boards are at right angles to the main board and cantilevered - an inherently weak arrangement and very vulnerable to accident. The stresses produced at the base of the pins tend to bend the board and break the traces. This mounting technique is cheap and neat - but a no no where vibration or shock are factors - or where boards are often removed and remounted. So if you anticipate trouble, plan on mounting the boards seperately and either wiring them permanently together or using plug and cable.

The greatest advantage of the SWTP kit lies in it being a complete system, with all that is needed for it to be useful. There are cheaper kits available, but they have less in them, and anything less can prove irritating. Those working on time share have found that a limited screen memory allows a mere glimpse of ones data - now you see it, now you don't — before it vanishes forever as the dumb thing at the other end of the phone line relentlessly regurgitates the contents of its output buffer - typically 300 characters.

The SWTP kit has an adequate memory, options for sophisticated cursor control and the ability for the memory to be used as an extension of your computer's.

But if you can't afford the goodies in the more expensive kits, or have a sufficently limited application that a simpler device ia adiquate, a TVT from THE DIGITAL GROUP, P.O.Box 6528, Denver, Co. 80206 may be just what you are looking for at about $100.

Regarding building TVT's from articles in hobby magazines - if you are an experienced builder, you can evaluate them for yourself. If not - don't



8008

The best 8008 based kit we have seen is the 008A. It has an unusual I/O arrangement for this chip, a 256 port bus. From:

RGS ELECTRONICS

3650 CHARLES ST SUITE K SANTA CLARA, CA 95050
(408) 247-0158

Another kit is the MIKE 2. We haven't seen one but hear good reports. From:

Martin Research Ltd., 1825 S. Halsted St. Chicargo, Illinois 60608

Report SPHERE

96 EAST 500 SOUTH - BOUNTIFUL, UTAH - 84010 (801) 295-1368

The first kit based on the Motorola 6800 chip has been eagerly awaited - and it looks like it will be the Sphere. We haven't seen one yet, but have talked with Michael Wise by phone. The differences between the Sphere and the Altair add up to a divergence of policies sufficient to add a new dimension to the expected 8080/6800 market battle.

The initial deal sounds incredible for the price: CPU, 1K PROM,4K dynamic RAM, 16 line x32 character module for home TV display, 73 key keyboard and the necessary power supply - all for $650.

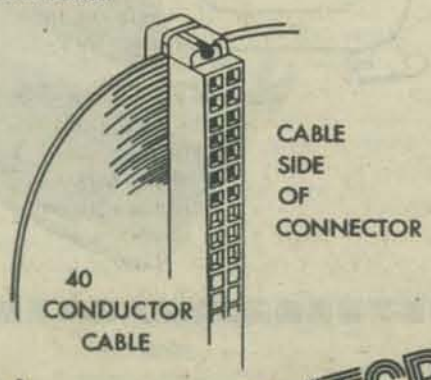
At first sight this looks a much better deal than the Altair, but the unit has no front panel, much of the cost of the latter. Sphere buyers will be up and running with a high level language for quite a bit less money than Altair owners, but they will have more difficulty if they want to work in assembly language. They will also miss one of the real advantages of the MITS system, the Altair front panel is really well arranged as an educational tool for teaching binary operations. Sphere BASIC is being developed to be very similar to HP BASIC in its instructions while MITS is close to DEC BASIC. This will become a significant factor in any choice.

Any new venture has problems with capitalization. Sphere has a super deal as an introductory offer, but are asking cash in advance. You save quite a bit of money because you trust them to deliver. Their price seems about right for what they are offering, a healthy sign, and what limited contact we have had with them left us with a good impression. For instance, they don't mind admitting that they are being carefull with both their funds and their promises. Pity more companies aren't the same way!

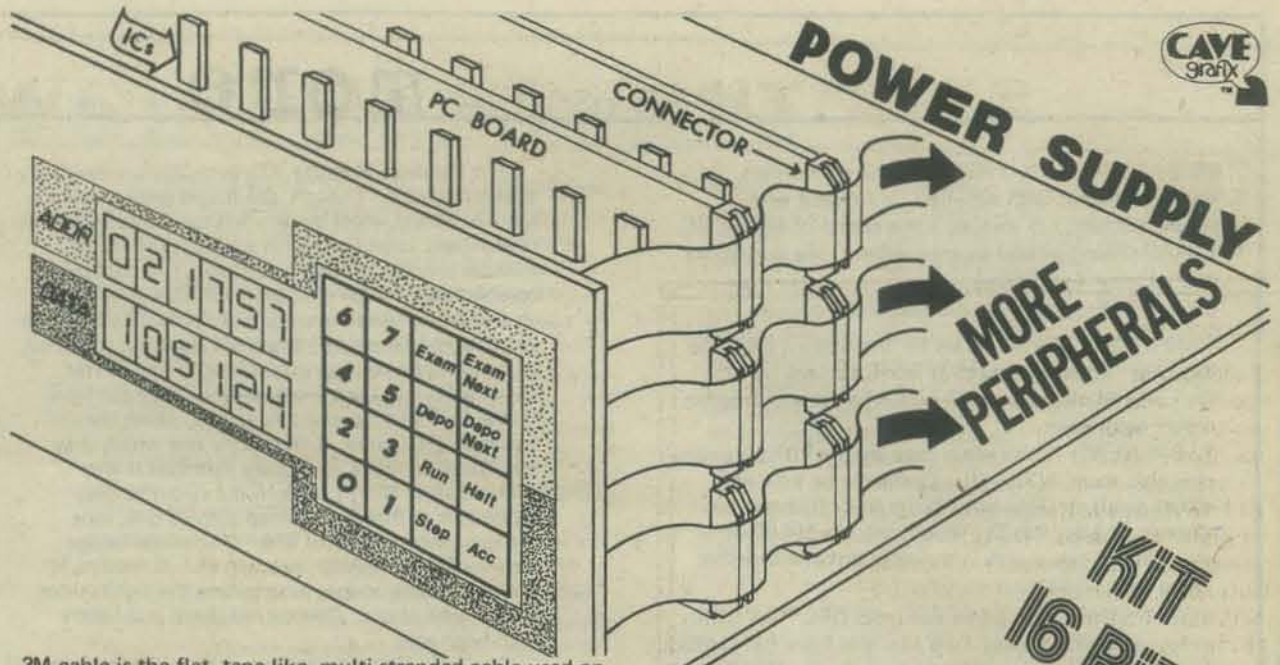
Report

Bill Godbout has been a mail order parts supplier for years, with an enviable reputation for producing good deals. (A 4K x8 bit memory board for $163 for instance.) His latest is the first 16 bit computer kit - a true minicomputer, not a blown up micro. And not content with this, he is about to produce the second!

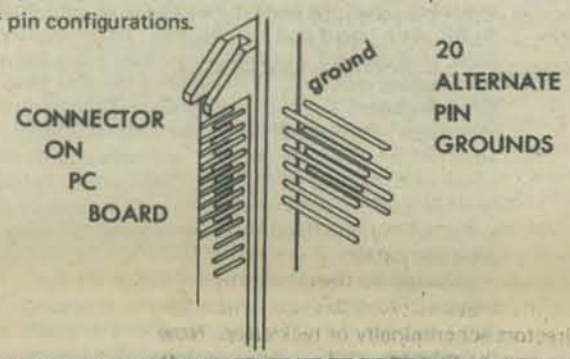
The first is based on an LSI chip which behaves like a NOVA CPU - four accumulators and a similar instruction set. To get the cost down, some corners have been cut and every unnecessary frill ruthlessly purged. Surprisingly, this has improved the beast. Rather than an LED front panel display, it has an octal numeric readout. The idea may have been to save money, but the result is both clarity and convenience. Another cost-cutting idea, the bus is 3M cable. This saves the expense of a motherboard and its sockets, while allowing a flexibility which will generate much envy in those who have to suffer with conventional busses.



MINI MICROCOMPUTER



3M cable is the flat, tape-like, multi stranded cable used on computers. Connectors are cheap, both the ones soldered to the PC board and those on the cable, and the latter are simply pressed into the wires. So if you want to build your machine into your desk, you can have a board in each drawer and a flexible bus between them! This may not sound much advantage, but it means you can take the whole bus out to a peripheral, or use boards of otherwise incompatible sizes or pin configurations.



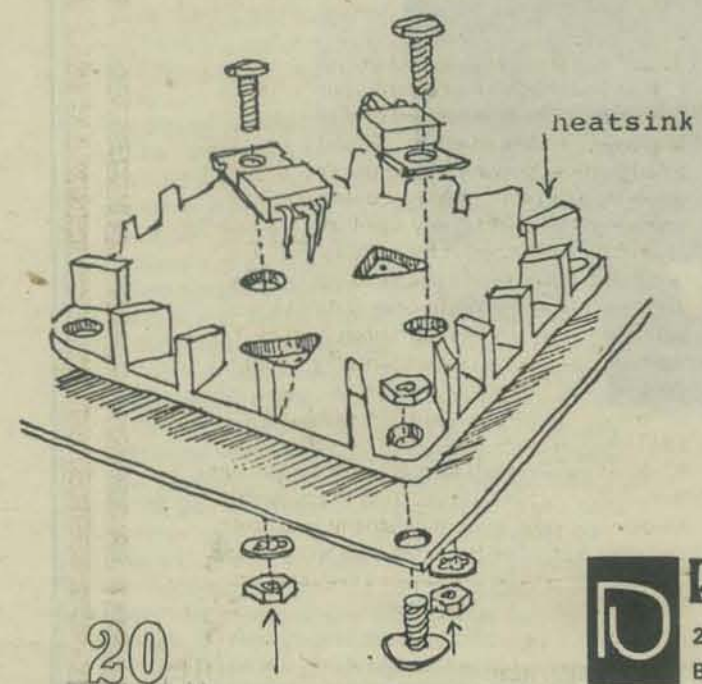
Godbout's second kit - for release mid October, is an MSI (Medium Scale Integration) copy of the PDP 11. It has a fast clock, 300nS, but is a micro coded device, so the effective speed is a lot slower, presumably, than the clock. The big questions are whether it uses the same instruction set as the PDP 11/40, and whether DEC allows the use of the Unibus concept. We will keep you informed!

GODBOUT

BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

Report

IT is common knowledge that MITS is having severe difficulties filling the thousands of orders thay have received. Don't blame them too much for the delays. They expected to sell 500 Altairs the first year, hoped for 1000, now find themselves having to build more than that each month. They were thrown in at the deep end when the first article appeared in *Popular Electronics* (we hear that it was printed two months before they were ready for it??) and they have not yet acheived control of the situation. The inevitable production delays, to quote from a comment heard at the Homebrew Computer Club, have spawned a whole new cottage industry. Boards designed to plug into the Altair bus are being made by the thousand. Will this hurt MITS? Not much, and it may even help them.



The largest question mark hanging over the Altair was "Is this a toy, produced in small quantity and then forgotten because it isn't supported?" The answer now is an emphatic no – principally because of the extent and quality of this second sourcing. The Altair made it big because it was a good design at the right price for the market concerned. It is now so widely supported that its bus is a de facto standard. If you have an Altair, you can be confident of support for years which makes buying the mainframe increasingly attractive.

Processor Technology is one of the companies producing plug compatible boards for the Altair. We saw some of their first assembled units and were impressed. Since then we have learnt a lot about them and put together two of their kits: a 4K static RAM board and a 2K PROM board. We've also seen the first production batch of I/O boards. They have three parallel ports and one serial, with program selectable Baud rates.

We had only the preliminary instructions but had no real problem assembling the kits. The boards are just beautiful! We have seen little professional equipment that is better. Gold plated contacts; milspec, high speed, low power consumption memory chips; all quality components and, lest people use it in desert heat, a heat sink of generous proportions and comforting ugliness. The kits were not just complete. They arrived with a coil of quality solder, soldering instructions, even wire for jumpers. And in neat ziploc packets at that.

Bob Marsh, who runs the company, explained how come the general quality level. Seems they designed the board, and by the time they got to production, all the components had dropped in price by 15%. So, rather than drop the announced price, they put the extra money into higher quality components, figuring that if they put out a super product they wouldn't have to deal with many complaints...

Processor Technology Co.

2465 Fourth Street
Berkeley, Ca. 94710
(415) 549-0857

The PAIA Gnome Micro-Synthesizer

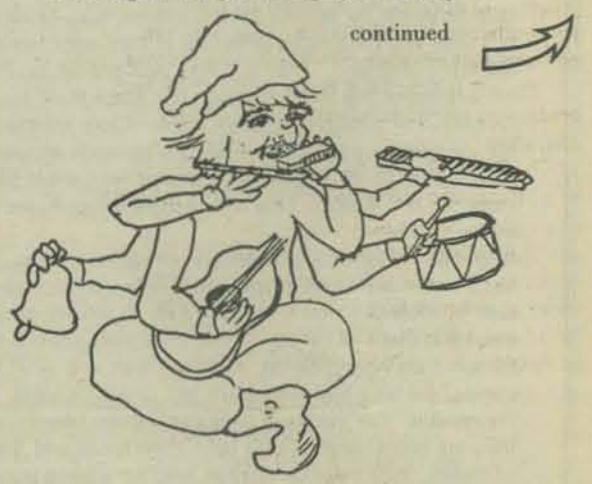
a review

by Howard A. Hurtt

I was assigned the task of evaluating the performance of the PAIA Gnome after successfully driving LeRoy Finkel out of PCC/Dymax with it. "Get that idiot and that damn box out of here," was how my assignment started. My interest in electronic music dates back to my early childhood, when I could entertain myself for hours with the sound of a Waring blender.

The Gnome is not a musical instrument, but it can help provide some insight into what musical sounds are made of and how they are synthesized. What puts the Gnome a class below a Moog is its inability to produce tones with repeatability. The tonal range of the micro-syntheiszer is continuous within a floating setting, and finding fixed notes with the slider is all but impossible. A mere tweak of the range knob is enough to send the whole scale bounding two or three octaves left or right. Songs can be played only by

continued



STARTING UP YOUR OWN CENTER

Once you have decided that your group can no longer muddle along in happy anarchy, you have to decide on the type of organization you wish to form, then find out how to go about it. If you buy the advice of a lawyer the probable set up cost will be about $1000. Can you do it yourself? Yes, read on...

Gnome plays on

" sneaking up" on notes which gives a Gnome concerto a distinctively sleazy quality. In a recent duet with my father, a professional harmonica player, I found myself sliding bumptiously up and down, chasing the 64-hole chromatic Hohner with the fervor of a rabid Chihuahua. The harmonica player soon began to weep and grunt and miss notes, which enabled me to catch up with him. We kept ourselves in stiches until a humorless spectator turned off my amplifier. One disadvantage of unconventional instruments is that some people automatically define different as bad.

The PAIA instrument has pretty fair versatility for something its size and price range (about $50 in kit form). I was able to more or less simulate a piano, a bell, a drum, a guitar, a flute and even a harmonica with it, to Dad's dismay. The Gnome's forte, of course, is sounding like a Gnome. Once one gets acquainted with the principles of fundamental wave shape, skew. timbre and envelope, almost any sound imaginable can be rpoduced. Matter of fact, give me a Gnome, an echo chamber, a multi-track recorder and some time, and I'll give Walter Carlos a run for his money. A good stout hi-fi amplifier is a must for proper operation of the Gnome.

There are a few problems with the PAIA Gnome which tend to make its operation something less than ideal. Most notably, the trigger pushbutton switch has lousy action. It is much too stiff and bouncy for comfortable, accurate noting. I would recommend replacement with a switch with cleaner action. Next is the problem of control interaction. Many of the controls, particularly those in the voltage controlled-filter section, tend to influence the output of the unit when they are not supposed to. To take the VCF completely out of action, it is necessary to turn it off at two locations and rotate four potentiometers fully counterclockwise. The VCF repeat function is very mysterious. It seems to work as expected only when it is inclined to do so. Finally, the controller range, which influences the tonal width of the slider, only works reasonably in the upper third of its rotation. The master oscillator range has the same problem. This seems to be a design error.

Despite these limitations, and its overall unpredictable qualities, the Gnome is an educational and very entertaining kit. It can do half as much as units costing ten times as much. It is easy to put together, having the most lucid assembly instructions I have ever seen on a kit. The completed synthesizer is functionally tidy and solid, which it needs to be, considering the force necessary to push the trigger button. The literature includes a very clear and informative tutorial on synthesizer theory and operation. In about an hour, if you bother to scan the instructions, you can be tailor-making your own noises. And if you are good at feeling your way around and faking notes, you can be butchering songs shortly afterward.

The PAIA Gnome would be ideally suited to a progressive secondary school music program or a college course in synthetic music. It is also well within the range of the average electronic music freak for zooop-ing along with Stevie Wonder at home. Now, if we could figure out a way to drive the thing with an Altair...

There are two basic types of organizations; partnerships and incorporations, and they may be profit making or non-profit. A partnership is a group of individuals joined together for a common purpose and recorded in the State records as such. Each partner may transact business for the partnership, and EACH PARTNER MAY BE INDIVIDUALLY HELD LIABLE FOR THE CONSEQUENCES OF THE TRANSACTION. So if you are a partner, someone else may incur a debt - without your knowledge or consent - and you may find your personal assets being seized to repay the debt. Partnerships have other problems. In some situations a majority is not enough. One no vote may act as a veto, so one partner may be able to paralyse the business activity. and the partner concerned cannot easily be removed. Before passing on to incorporations the author has to confess some bias. He was once in a partnership.

Corporations are an English invention to deal with the problems of partnerships, and their principle purpose is to limit the liability of those involved. In the eyes of the law, a corporation is a person (lawyers are sharp and have no difficulty distinguishing between a natural person and a corporation). This person is, not being human, comparatively incompetent and so its affairs are controlled by a board of disinterested persons called directors. Since these worthy people are merely giving the person advice, any debt incurred is incurred by the corporation - not the directors. And this is so unless the directors act criminally or recklessly. Now you may be thinking, "Whoa there, back up. What was that about the directors being disinterested?" Well, that was really the cute part about this strange English invention. The directors are disinterested because they don't own the corporation. They are merely good hearted fellows willing to be appointed to run it by the true owners. These are people who bought shares in the corporation. They own it, but take no part in running it - except that annually they elect the board of directors. So the directors aren't liable for debts because they don't own it, and the owners aren't liable for debts because they don't run it. (And the true genius of the inventors becomes apparent when you realize that there is no reason why the shareholders can't elect themselves directors.)

In short, if you form a corporation, all you lose if it goes bankrupt is the value of your stock. This is nice but there are other advantages. Decisions are made by a simple majority vote, so business goes on regardless of arguments death or incapacity of individuals. There are also clear rules governing the conduct of the board which protect the interests of minority stockholders. But the reason for the clarity of such rules is that there is a bad side to corporations. Corporations aren't vulnerable to disruption by individuals with honest differences - but are susceptible to manipulation by the sharp operator, even in the absence of outright fraud. Courts have rightly taken a jaundiced view of the behavior of such organizations and endeavored to lay down rules to prevent manipulation - but with limited success. Nevertheless, by far the most common form of business organization is the corporation which is testimony to overall advantages.

So far we have only talked in terms of normal businesses - organizations formed to make a profit for the people who own them. There are also non-profit organizations. The name is unfortunate and misleading, because they are fundamentally different from normal businesses and the difference has nothing to do with profit or loss - IT IS A MATTER OF WHO OWNS THEM.

Who does own them? Well as a matter of fact, you do.

All you people are members of the public and the fundamental distinguishing feature of a tax exempt organization is public ownership. Thus the true nature of a non-profit, tax exempt organization is that of a charitable trust (from which they are derived). That is to say an organization which controls and uses public funds for public good.

Now the fact that the public owns it does not give the public the right to directly control it - so tax exempt organizations are known as being quasi-public in nature, and they fill a curious niche exasperating to both the Judiciary and the Taxman. The judiciary finds great difficulty in penalizing the owners - the public - and the taxman has the same problem when he wants money. So what both do is try to limit tax exempt status to organizations which are genuinely public in nature, keep a close eye on them, and punish severely any failure to observe strict rules of conduct which they have laid down.

What sort of organization should your group form? Well - for starters, take my advice and plan on incorporating. Whether to go non-profit is more questionable. If you own your corporation, it's yours. If it makes a bundle, you must pay tax on it, but what's left you may put in your own pocket. If it goes broke, you lose the value of the stock you bought, but once the creditors have been paid, you keep what ever is left. Unless you sell your shares or it goes broke, you can't claim tax credit on money you put into it, because your only claim would be for a business loss.

A tax exempt type of organization is quite different. If you put anything into it, it is gone forever - since you have given it to the public, and the public is not about to give it back. (If your organization collapses, any remaining assets must be given to some other non-profit organization.) However, since you are giving something to the public, you can immediately deduct the amount from your taxable income. Even the IRS doesn't feel it reasonable to tax you a percentage of what you gave totally to the public. If your organization makes a bundle, the taxman won't ask for a penny - but you can't put it in your pocket because it belongs to the public, not to you. The ONLY thing you can do with the money is to spend it for the purposes for which your organization was formed. And to make sure that you do, both the judiciary and the taxman will require much higher standards of conduct from you than they would expect if you were profit making. Rightly so, because its not your money to be careless with.

So what's the point in being non-profit? The point is this. If you have a project which you feel is worthwhile, which will benefit the public, you can do your thing, tax free. And if you make a bundle, you get to direct the spending of it. And if there are public funds available, they can be given to you to spend - which can't be done if you are in business for yourself.

Now in either kind of organization, you can be paid for what you do - and paid enough to properly recompense you. So don't think that a non-profit organization need pay any different wages than its profit making counterpart. The taxman will be alert for any sign that a tax exempt organization is being run for personal gain, but certainly doesn't expect people to starve. The people who run non-profit organizations are public servants, just like the bureaucrats in Washington, and the taxman (unfortunately) doesn't mind the salaries they get.

How much will incorporation cost? About $25 if you do it yourself. How do you set about it? What do you need to qualify for tax exempt status? Read the next issue. And if you want a model non-profit accounting scheme, read the one after that. And if you want all this garbage, with sample (PCC's) By-Laws and Articles, send us $2 for a xeroxed booklet.

by Keith Britton

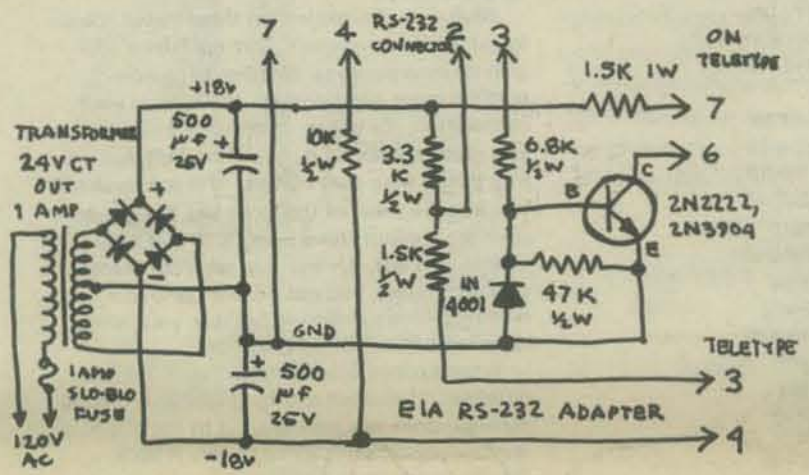
CURRENT CONCERNS

by LEE FELSENSTEIN

The Teletype Model 33 will be around for a long time. In spite of its noise, slowness, maintenance problems and slow delivery, there is nothing equivalent at twice the price.

The Teletype Corporation has some very good manuals available which explain the operation and adjustment of the mechanical innards, but the crucial task of connecting the thing up to the outside world is not covered by Teletype or anyone else. In fact, those of us who do that sort of thing agree that the knowledge of how to hook up a Teletype is folklore, passed on by oral tradition!

So, for those of you who are staring at your new (or used) Teletype and are wondering where the data plug is, this article will reveal to you the secrets of the MYSTERIOUS AND OBSCURE CURRENT LOOP!!



SOME HISTORY

The teleprinter (now known as a Teletype) was invented around 1917 as a high-speed printing telegraph sender and receiver. It was designed to connect with a lot of other teleprinters through a single loop of wire.

On a railroad telegraph system, for instance, the Teletype in each station would have a wire coming in from up the line and one going out down the line. At the end there would be a 48 volt battery and the current would be returned through the rails to the other end of the line. So there was a loop which current would flow through when nothing interrupted it.

Inside each Teletype there were two sections; the sending part and the receiving part. The sending section was basically a switch which was normally closed. The receiving part was an electromagnet which would start the printing mechanism when it let go.

The two sections were connected in series. When nobody on the line was sending, current would flow in, through the sending switch (or distributor), through the electromagnet (or selector magnet) and out to the next machine.

When someone started typing, the distributor in their machine would open and close in a carefully-timed sequence as they hit each key. When the switch first opened, the selector magnet in all of the Teletypes on the line would let go and the printers would start clattering. The data was transmitted into the mechanism of the Teletypes by the motions of the selector magnets as the printers went through their cycles.

In telegraph talk, a line with current flowing in it was said to be in a "marking" condition. A line with no current was said to be "spacing". These terms are still in use, but digital electronics has added numbers to the game. A "mark" is equivalent to a '1', and a "space" is equivalent to a '0'.

The railroad-type current loop hookup (called 'simplex connection') is no longer in use, but Teletypes are still built as if it were being hooked up that way.

WHAT'S INSIDE

When you get the cover off your Teletype you will see the Electrical Service Unit on the right-hand side. Lots and lots of wires.

All of these wires end up at a panel of square white plastic connectors on the back. Don't worry about them. Look underneath them on the rear panel and you will see a black plastic "barrier strip" with screw terminals for connections.

You may have to remove a grey fiber insulating strip to see it. Before monkeying around in there UNPLUG THE POWER CORD! The two left-hand terminals on the strip connect directly to the power line. You could get a very big surprise if you touched them while they're "live".

The fiber insulating strip has numbers embossed on it. these are the numbers of the terminals. They start at the left and go from 1 to 9.

Terminals 3 and 4 are the distributor contacts. The contact is made by a carbon brush and the resistance is current-sensitive. An ohm-meter will show a few hundred ohms resistance, but with 10 milliamperes of current going through the distributor the resistance is less.

Terminals 6 and 7 are the selector magnet driver inputs. The driver is a transistorized circuit on a green card in the Electrical Service Unit. The input to this circuit is sensitive to polarity - pin 7 must be positive with respect to pin 6. Only when current is flowing in that direction will the Teletype sense a "marking" condition.

You can tell when this happens because the Teletype will stop chattering when its switch is in "line" position.

GOING OUTSIDE

When hooking up the Teletype as a current loop device, there are no standards as to which pin of the modem or computer interface connector goes where.

Here some knowledge and documentation about the modem or interface is necessary. A few general notes are possible, however;

* The distributor contacts (3 and 4) usually connect in series with a terminal in the input section and a terminal connected through a resistor to a voltage source. This resistor should be able to provide 10 milliamperes to the contacts.

*As mentioned above, current flowing in either circuit represents a binary "1". Current should be flowing when no data is being sent.

*In a modem, the higher-pitched of the two tones represents a "mark" or a "1".

*The selector magnet driver circuit drops only a few tenths of a volt with its rated 20 milliamperes flowing through it. It should not be connected directly between a voltage source and ground without a current - limiting resistor in series.

RS - 232, ANYONE?

There are standard interconnections if another type of hookup is used. The Electronic Industries Association (EIA) has set up standard RS-232 for data connections.

RS-232 uses positive and negative voltage levels to represent "mark" and "space". A device using this setup cannot be hooked directly to the Teletype, but needs a converter. A schematic diagram of such a converter is shown. It's simple to build, but watch out for the AC power line.

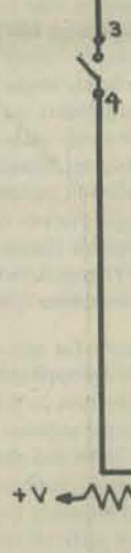
The importance of RS-232 is that a Teletype which is adapted for it can plug into any other piece of equipment which is set up for it.

In RS-232, a negative voltage level of more than 3 volts represents a 'mark' or a '1', and a positive voltage level of more than 3 volts represents a 'space' or '0'. The load presented to it must be greater than 3000 ohms and the transmitting side must be short-circuit protected.

A standard "dataphone" or "DB-25P" connector is used. The plug used has pins and the connector on the modem or interface has sockets.

Pin 2 is the data from the Teletype to the modem or interface
Pin 3 is data from the modem or interface to the Teletype
Pin 7 is ground
Pin 4 is the request-to-send signal, which may be necessary to enable the modem. "Mark" turns this signal "on".

Take care in soldering the connector, as the nylon insulation tends to melt with too much heat.



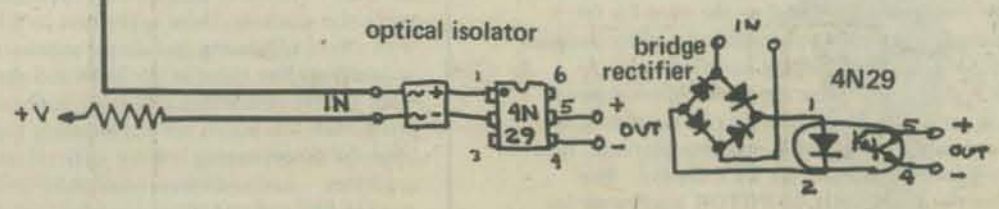
MORE, MORE, MORE, AND NOW!

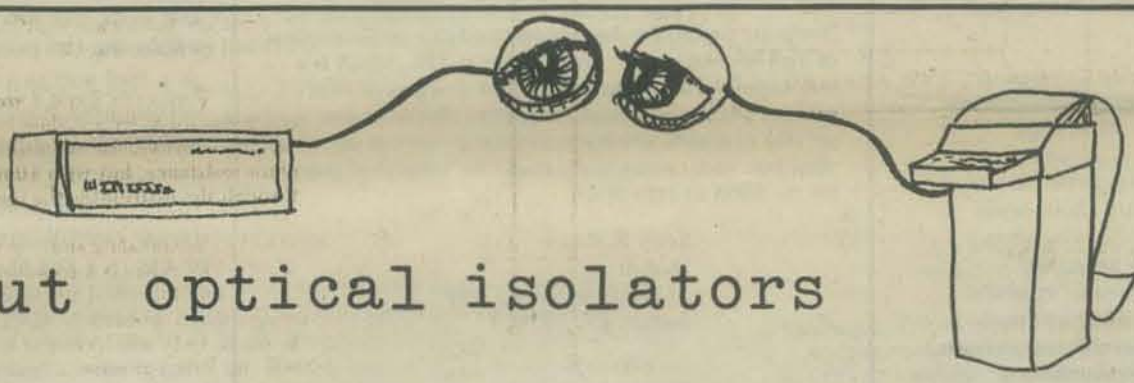
Many hobbyists seem to want to connect a Teletype, a modem, and a microcomputer interface at the same time, so as to put their micro "on line" as a kind of intelligent terminal.

This is difficult, because the modem and interface designers have made the assumption that theirs will be the only device hooked up to the Teletype. They designed their circuits so that one side of the distributor and one side of the selector magnet driver are connected to fixed voltages in their equipment. You can't take two such pieces of equipment and hook everything in series.

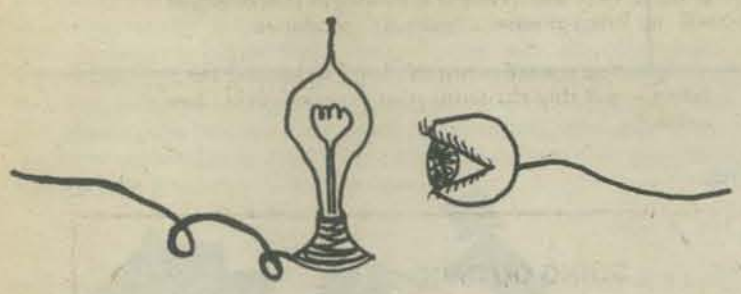
A little circuit you can build will provide the necessary isolation between current-loop devices. It consists of a 4N29 optical isolator and four diodes or a diode bridge rectifier. It drops about 3 volts across its input terminals, so you will have to have more than 6 volts available to run two of them. Nothing comes free.

The isolator is not sensitive to polarity on its input , though.

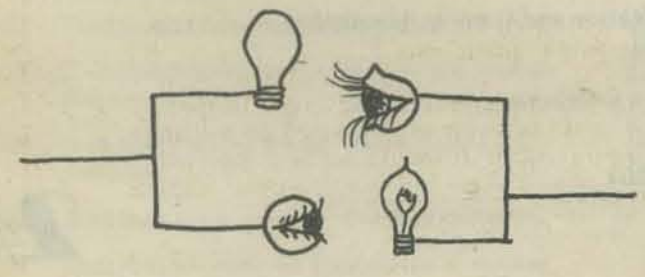




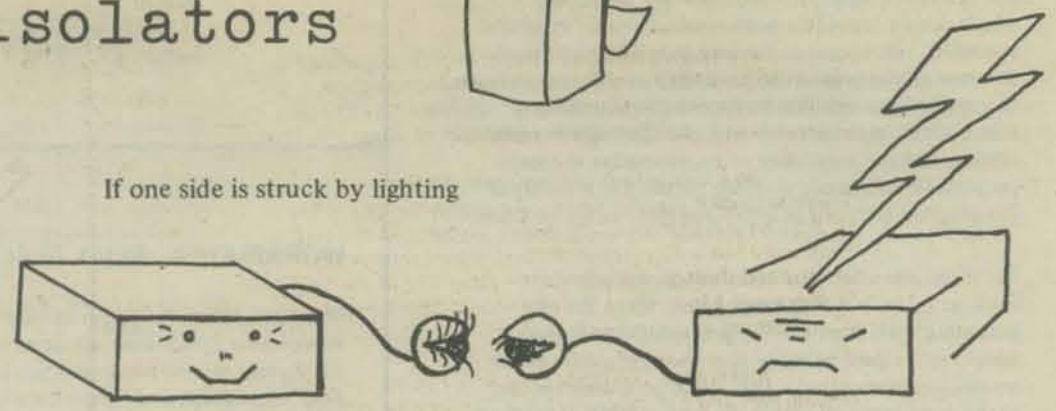
more about optical isolators



An optical isolator is a device made up of an LED (light emitting diode) pointed at a photo transistor. The current passing through the LED controls the light it emits, which is the light received by the photo transistor, which controls the amount of current the photo transistor allows to pass. So the current in the first curcuit controls the current in the second circuit, but the two circuits aren't connected! Now if you have two of those - pointed in opposite directions - it looks like a simple continuous piece of wire to both sides... but...

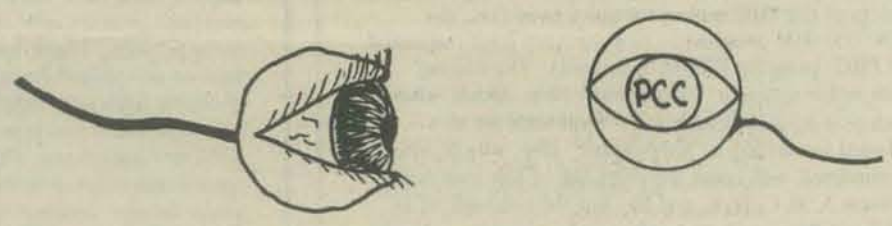


If one side is struck by lighting

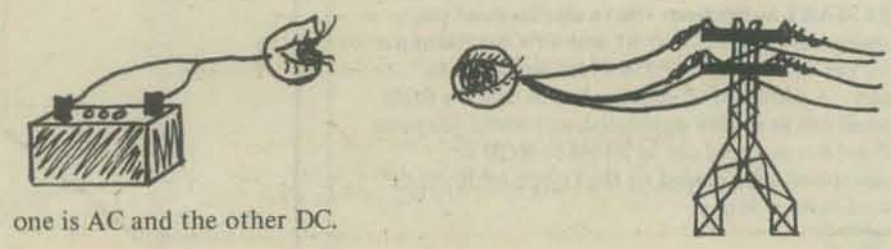


the other side doesn't mind a bit and if...

one side has different voltages and/or current



both think the other is wonderful... even if...



one is AC and the other DC.

MACHINE OR ASSEMBLY?

This letter continues the discussion started in PCC Newsletter Vol. 3 No. 4, on the desirability of assembly language. The fact that some discussion exists is most encouraging, as it indicates that the question was properly raised in print. Naturally, each of us has personal concepts that he champions, concepts formulated from the perhaps incomplete data which we have had available. Public advocacy of a particular system is in our good interests, that we may gain from the discussion a new point of view. But to work for us, this advocacy should be regarded intellectually, that we may discuss rationally, avoiding emotionalism.

I advocate the use of octal machine-language programming on small 8-bit computers. My previously published arguments were generally based on the need for far greater memory to store an alphabetic assembly program than is required for the resulting machine code. As Mr. French points out, these arguments break down if the system translates each assembly command *as it is entered*, stores the command in machine-language, then translates it into alphabetics for TTY display. The system discussed is the MIL MONITOR 8 software for the 8008; this is a good starting point for discussion of the program entry and editing required in any small computer system.

Fundamental to the concept of assembly language is the knowledge that each alphabetic command represents a single machine-language instruction. Most large-computer assemblers are "symbolic," allowing an instruction-location to be tagged with a particular name. (There is some confusion in the industry about describing assemblers. In his *Minicomputers for Engineers and Scientists,* Korn differentiates an "assembler" which simply translates mnemonics, from a "symbolic assembler" that permits the user to refer to addresses in terms of symbols. I use his definition for convenience.) In a symbolic assembler, jumps are programmed by name, rather than memory address. This allows easier editing, and the possibility of re-assembling the same program at a different absolute location in memory if the program is stored in bulky assembly language form.

But if the assembler *just* translates alphabetics into machine code, is it that much help? Since the programming itself is an off-line process; short, logical, tables can be used to easily find the octal codes which are not yet memorized. And, occasional direct access to the octal instruction may be useful; 8008 redundant codes can provide useful software flags. But, assuming that you would still rather program in letters, how much more will it cost?

First of all, the MIL ROM's are no longer available, but this discussion is on efficiency, not availability. Some aspects of the MIL system are quite good (i.e., the TRN: translate program to new memory page command, and PRG: program PROM command). The editing (such as for setting a "break-point" step, which, when (such as it is) is in octal only. Commands are also included for setting a "break-point" step, which, when encountered, will cause the print-out of the contents of registers A, B, C, H, L, and M. But the contents of D and E are destroyed, and only one breakpoint at a time is allowed. Thus, the MIL breakpoint system is not the highly-desirable machine-language trace.

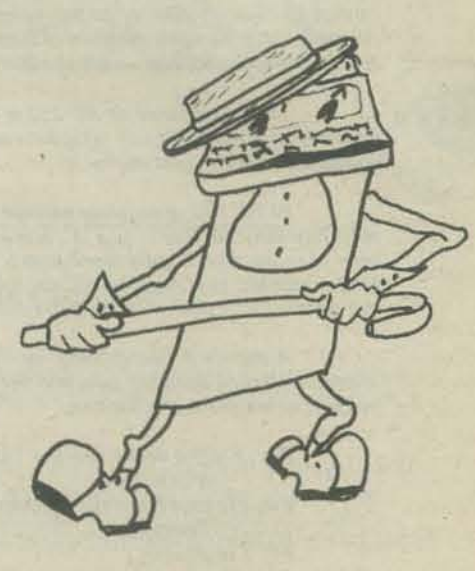
The MIL system utilizes over 1.5k of ROM, and *all* of the RESTART instructions. But a simpler octal programming, editing (such as it is), and TTY octal dump system can be contained in 256 bytes, anywhere in memory. A whole lot of memory is thus saved, a ROM this small can be readily duplicated, and useful programs which are not included can be added to ROM or perhaps stored until needed on the *Computer Hobbyist* standard cassette format.

I keep writing "such as it is" about these editors since I feel that the computer should be able to create space for forgotten machine-language steps by "bubbling-up" subsequent steps *and correcting the jump addresses;* it should similarly be able to delete steps. This is somewhat complex, not all that bad, but if this is already running on an 8008, I am not aware of it. Presently, the prudent programmer will insert NO-OP's to save some room for future modification and expansion. Similarly, an ideal system would include CRT editing and full CRT trace of each register after every machine-language step, or as desired. This system is feasible on the 8008, although extra hardware is needed.

I should note that the arguments for and against an assembler should be tempered by application to a particular machine; these apply best to 8-bit machines with direct addressing and simple register-transfer instructions like those in the 8008 and the 8080. Somewhat greater instruction efficiency is obtained in instruction sets which are less humanly logical; machine-language programming is more difficult on these machines. And some larger-computer instruction sets (16 bits and up) are so complex that they virtually require the use of some sort of symbolic assembler.

Nor is my position to be misconstrued as a condemnation of high-level languages; my true love is APL, which is a very high-level language indeed. But in terms of what you pay (data storage and/or memory) for what you get (the ease with which you make the machine do what you want) assembler is about the worst deal going for the 8008 and the 8080.

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

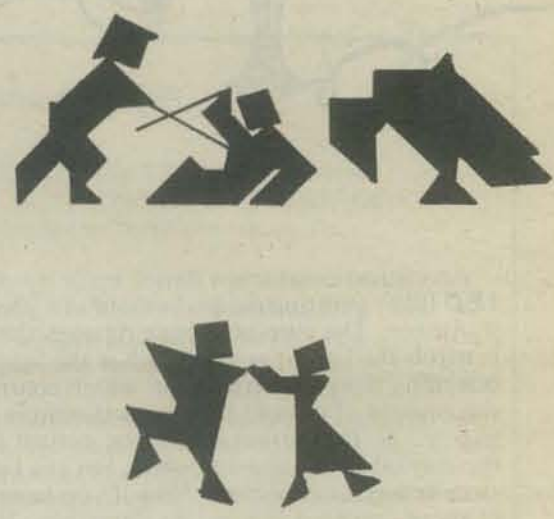


COMPUTER PEOPLE – The Boston People's Computer Collective is a group of individuals interested in increasing laymen's awareness and understanding of the role of the computer in our lives. We are planning to offer courses exploring the possibilities of the computer, both as a tool and as a toy. We are also investigating setting up a traveling hands-on interactive computer exhibit that would appear in local schools and shopping centers. If you have any time, energy or ideas that you would like to share, please call Bill Mayhew, 617-522-4800, X25, 9 AM to 5 PM Monday through Friday.

WORKSHOPS SKILL BUILDING PAPERS

The 1975 North American Simulation and Gaming Association Conference will attempt to accommodate the variety of interests in the gaming and simulation field. Workshops and formal papers will be presented on design, application, theory and research. Experienced resource people will be available to help conference attendees deal with the problems of conducting and designing games/simulations.

The conference will focus on communication and participation, with many opportunities for attendees to meet and interact with each other in game and non-game settings. Those making presentations at the conference will have the option of reading formal papers or choosing from such other formats as gaming/simulation exercises, multi-media presentations, informal discussions and workshops. Papers are encouraged from all contributors including those persons making non-formal presentations; they will be used as a basis for discussions and will be published in the formal proceedings.



North American Simulation and Gaming Association
c/o COMEX PROJECT
University of Southern California
University Park
Los Angeles, CA. 90007

SUTHERLAND'S HELMET

TO: Fellow Freakie Fantasists
FR: The Fantom

A Piece Of Paper On The Possibility Of Using the HELMET Display System In Computer Games And Simulations

STEP 1... Get *Computer Lib* by Theodor H. Nelson; read the *Dream Machines* side. If you are currently poor, look at page 9 in the Jan. 75 PCC; read, especially the bottom right corner.

THINK . . . what could you do with a 3D perspective display????? Think . . . what is the neatest use computers do today????? think . . . PCC . . . peoples . . . aha!! you've got it ?? what is it ??? games . . . right you are. Sutherland's Incredible Helmet offers a degree of unrealism (or was that realism???*) never available before . . . a tremendous degree of interactivity . . .

START withsome staid, mouldy oldies of the computer game world, LUNER LANDER you say?? that's a good start . . . build a LEM. . . a Sphere for a body, paint the outside black. . . on the inside add standard spaceship stuff – life support, radar, pilot's chair. . . add weapons control and you have a warship . . . look out Captain Kirk – make a spacesuit for the pilot. . .use Sutherland's Helmet for his helmet... drive the whole thing with an Altair??? add a real-time perspective unit . . . from this one unit you can conquer space . . . Lander, Spacewar, Docking games . . . figure out software to make a moon, take the lander out for a cruise. . . landing on the moon – see the craters rush up at you . . . keep watching the radarscope for altitude . . . steer your lander into that large crater to the right . . . perfect landing??? watch the dust slowly rise, then fall back to the moon . . . take off again and try to dock with the command module . . . graphics on the same level as *2001* . . . too bad that we can't control gravity . . . design some kind of system to simulate space walks . . . the possibilities are endless – use this toy to teach relativity; as you approach the speed of light watch the stars red-shift – whoops . . . I added color – anything's possible.

Build a pair of tanks ??? how about a race car game? submarines ?? what about taking the airplane simulators one step farther by adding visuals ?? this is already being done . . .

ALAS AND ALACK
THERE IS A DRAWBACK . . .

$$$$$$$ and time and equipment and . . . there is always tomorrow???

In the meantime, there is a need for more stuff to make more games possible . . . one of the things needed most is a good joystick . . . with a joystick and a good display such as the one featured in the *Computer Hobbyist* (see the last line) we can do anything that Atari can doo . . . walk into your local game parlor and look around . . . those things with the coin boxes stuck on them are computers, and look at all the different games . . . baseball, race games (GranTrak 40), Tank, mazes, Space race . . . all of these games can be made on a computer with a joystick.

How about a good, cheap plotter . . . art freaks abound . . . let's put TV our of business . . . everybody run out and start your own PCC. enlist your local Ham Radio operators, a large body of talent waiting to be tapped . . . Hardware isn't hard . . . grab you a teacher . . . this hobby is just starting and it is very hard to regulate it to death . . .

(LAST LINE. The Computer Hobbyist, Box 295, Cary, North Carolina 27511)
DOWN WITH BOTCH COMPUTERS – UP WITH INTERACTIVITY.
COMPUTERS WERE NOT RAISED TO TALK FORTRAN AND COBOL

Build your own Star Trek and Spacewar games for fame and fortune. It is not very hard – Rule 1. Plan exactly what you want to include in the game before you begin. 2. Allocate all variables logically, ex. E is energy 3. If you expect it to be very complex, plan it as several programs that chain. 4. Do everything in subroutines. 5. 'stack' variables to conserve memory space. 6. allocate loads of time and patience to this task. 7. Use your imagination. These are some basic and obvious rules, but they are not absolute. In making a space game, there are 2 paths that are usually followed; (a) make it following the Star Trek programs Enterprise, following its design exactly, making the program very realistic. (b) Make it with anything you want to put in it. It is often a good idea to look at a well organized SpaceWar game to see how it was done. Make your own Star Trader or exploration or colonization game. These games are more difficult than a Spacewar game, but allow you a greater freedom of what you want it to do.

John A. McClenny – 5819 Brenda, San Antonio, TX
78240

Any comments would be appreciated. P.S. I am also working with a robot.

SHORT STAR TREK ANYONE?

I am president of the Homes Computer Club, Holmes High School, 6500 Ingram Rd., San Antonio, TX. 78238.

We use an HP2000/F computer, programmed for BASIC. Most of the members of the club are programming in BASIC, but a few, via the system's translators, are into COBOL and FORTRAN 4.

We are greatly in need of a shorter Star Trek program that has the basic format of HP's $STTR1, but doesn't take up 8000 some odd words of storage.

Chris Moseley
Holmes Computer Club
1927 Harpers Ferry
San Antonio, TX. 78245

10 COMMANDMENTS

10 BASIC is thy language. Thou shalt have no other languages before it.

20 Thou shalt not make thy loops infinite, nor return with no prior gosub.

30 Thou shalt not take the name of Hewlett-Packard in vain.

40 Remember the integrated circuits, to keep them holy.

50 Honor thy programming instructor, tho feeble-minded he may be.

60 Thou shalt not kil-other's programs.

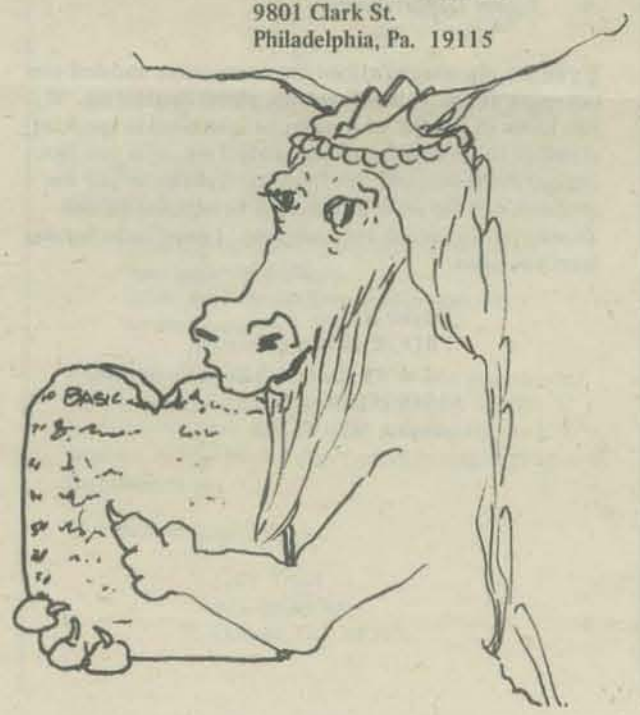
70 Thou shalt not explore the insides of thy terminal with thy fingers.

80 Thou shalt not steal another program and call it your own.

90 Thou shalt not sabotage thy neighbor's programs.

100 Thou shalt not crash the Computer or in any way damage its hardware or software.

Robert "moses" Zeidman
9801 Clark St.
Philadelphia, Pa. 19115



STAR TREK and SICREC and . . .

You may remember me as the fellow who sent the letter asking if anyone wanted to trade Star Trek games. The response was slow, but fairly steady for several weeks. I was somewhat disappointed, however . . . I guess I overestimated how many active computer-oriented Trekkies there were. Somewhat surprisingly, most of the answerers were high school people. Made me feel kinda funny, being an old man of 28. Man, I'm American Graffitti revisited!

Got my copy of WTDAYHR (*What to do After you Hit Return*) and think it is very good, in general, Would have liked to see more listings, but I suppose there were problems with copyrights.

At any rate, to tell you a bit more about myself (there's a reason, later on), I got my MS in computer science in December, 74, with an option in operating systems and compiler theory. My undergraduate degree was in geography (that's right! geography!); I am now looking for a school at which to continue my PhD. I decided that I didn't really like my school at Lafayette, La, so I'm going to look at some various departments this summer, although most likely Texas A & M (ever heard of Aggie jokes? . . . something like Polish jokes). I was president of the student chapter of the ACM at the University of Texas at Arlington, the vice-president of Upsilon Pi Epsilon (National honorary society for Computer Sci) at UTA, and again last semester at USL. I turned down a nomination for the ACM at USL.

I told you all that (in a modest way, of course, aw schucks) so you might consider this: I'm considering trying to start a new subgroup in the ACM: SICREC, special interest committee on recreation and entertainment uses of computers. SICFUN!

SICREC could have several objectives: to provide a convenient games exchange center – to help legitimize the use of games since ACM has some 26,000 members of the computing community – to help hold down the cost of game software in the face of the ridiculous price spiral seen in software by the industry in the past several years – to promote the use of various higher level languages in games (I personally don't have anything against BASIC, but in a large system, where it's available APL is preferable, I feel (no, I'm not an APL freak). Home computing is an idea whose time has come. The introduction of hardware, typified by the Altair 8800 will lead to changes in the home environment of an unimaginable nature and incredible extent which would have been pure fantasy only 5 years ago. However, it's nice, once you have the system together, to be able to do something with it, especially as a learning process, and I feel that games are ideally suited.

I can see several problems with copyright, and so forth, since there are only so many basic game structures, and infinite elaborations on these foundations.

Have you heard that there is a new magazine called BYTE slated for publication in August. It's from the publishers of 73 Magazine, the amateur radio publication. (I'm a ham, too – WB5KXH). And, of course, Creative Computing had appeared on the scene, so it appears there may be a scramble for this type of software. I definitely wish to keep hobby software at a friendly exchange level (I've heard of a Star Trek in the East that was being sold for $100 . . . I was AGHAST!), if at all possible.

At any rate, any suggestions you might have would be most appreciated. A few notes on other things:

I thought the story of the birth of the dragon in the last issue was most enjoyable, although I was rather disappointed to hear that you left Denver because of the industrial climate there. I love Colorado, and presently have a resume in at NCAR, in Boulder. I'm not firmly committed to a PhD at the moment, as it would be nice to go out and earn some money, but I really would like to get into something like PCC. Guess that California is several years ahead of the rest of the country, as usual. Sure wish there was something like it in Dallas, or Denver.

Robert R. Wier
1208 Mistletoe Drive
Fort Worth, TX 76110

TALKERS WANTED

As a student representative on the Student Affairs Committee, for ACM 1975, I would like to take this opportunity to ask you for some help in finding speakers for our Student Seminars. The Association For Computing Machinery (ACM) 1975 convention will be held in the Twin Cities on October 20-22, 1975. I hope you already know this and are planning to attend.

The Student Program for ACM '75 will consist of Student Paper Competition; Miscellaneous Activities, i.e., Computer Art, Computer Chess, etc.; and Student Seminars. We need people to commit themselves for speaking at these four forty-five minute sessions:

1. Historical Review of Standardization
2. Current Issues
3. Educational Opportunities
4. Career Opportunities.

If you are planning to attend the convention, and feel you can speak at one of these sessions, please contact me. If you know of anyone who might be interested in speaking at one of these sessions, please contact me, or if you like, contact them yourself. No fund are available to pay the speakers, and the sessions must not be commercialized. Thank you very much for your time. I hope to be hearing from you soon.

Wayne R. Asp
PROGRES Representative
ACM '75 Student Affiars Committee
1558 10th Ave. So.
Anoka, MN. 55303

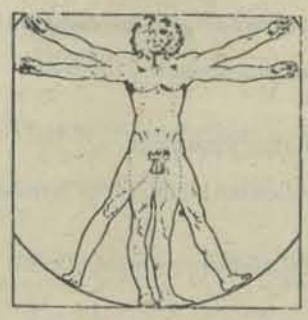
SOUTHEAST MINNESOTA AMATEUR COMPUTER CLUB

Three of us started a computer club in January by purchasing the Altair 8800 kit. We have grown to eight members just by word of mouth. We don't intend to advertise ourselves until we have a working system. While we have 4K of memory, a parallel I/O interface, an extender board, an audio tape interface, and edge connectors on order, we only have the Altair computer and 256 bytes of memory in our hands.

The kit went together easily and worked on the first try. We built it slowly with at least two members in attendance at each session. It didn't seem any more difficult to me than an audio signal generator kit from Heathkit I built a couple of years ago. We are generally satisfied with the quality of the components. There were only a few minor problems. Specifically – a few things didn't arrive with the kit and were listed in a cover letter, but the missing parts did arrive within two weeks; two of the ICs included were wrong, the first we caught ourselves and MITS called to tell us about the second before we got to it in the construction, both ICs were exchanged by MITS in about two weeks.

We don't know just what our final organization will be or just what all we will do with the computer, but things are happening too fast to plan very far ahead. We see four major applications areas: recreational, educational, personal business and household. My own interests are primarily in the first two areas as it appears so it is with you as well. The personal business applications could be anything from a check balancing program to a stock analysis and charting program. My wife has already asked me about two household applications – menu planning and inventory control for the pantry. Each of us in the club intends or hopes to get his own computer, but will want to remain in the club to use the specialized or occasional use of equipment and services we expect to offer thru the club. We expect to continue to own a club computer (or computers!) after we begin to get our individual ones. Quantity buying is another good example that the club could offer. For instance it looks as though we might be purchasing from three to five or possibly more computer systems at a time allowing the members to benefit from a possible quantity discount. We expect to add a PROM when our finances permit, which we could all use to good advantage. Certainly the club computer will be a relatively powerful one allowing members to start with a truly minimal system and use the club computer for advanced functions such as assembly, compiles, and hard copy printout – the last assumes that the user has no printing device or too slow or small a one for what he has to do. We have already found it most reassuring to find so many other friendly helpful people, expecially with skills and talents we lack. The best analogy so far to what is happening in amateur computers is amateur radio. However, the function of such clubs in the amateur computer field has not been appreciated until now.

Daniel Nicholson, President
SEMACC
2122 NW 17 Avenue
Rochester, MN 55901



"I'd like a computer that's about this high and this wide..."

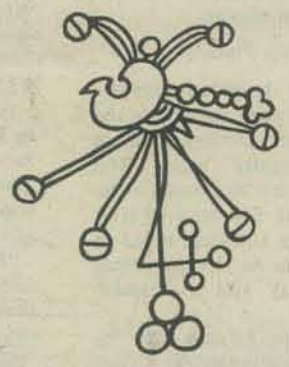
The Mits-Mobile is a camper van completely equipped with an Altair BASIC language system. Included is an Altair Computer, Comter terminal, ASR-33 Teletype, Altair Line Printer, Altair Floppy Disc and BASIC language.

If you want to find out when the Mits-Mobile will arrive in your hamlet, write to Mits-Mobile, 6328 Linn NE, Albuquerque, NM 87108 or call (505) 265-7553.

A nice poster titled "lucid Dimensions" by Bay Area illustrator, Clifford Spohn is available and free from:

Advertising
Memorex Corporation
San Tomas at Central Expressway
Santa Clara, Ca. 95052

They also throw in a couple of Memorex posters.



Program Style, Design, Efficiency, Debugging and Testing
by Dennie Van Tassel Prentice-Hall $10.95

This is a good book for those with some programming background who wish to improve their skill. The book is chock-full of hints on the topics in the title. Finally, it has one of the largest selection of programming problems available in any book.

James Douglas
531 Easterby
Sausalito, Ca. 94969

MID-MICHIGAN MICRO USERS GROUP

I'll be going to Michigan Technological University this Sept. and will send you the mailing address when I get it. I would appreciate any MTU computer freaks writing me, thanks.

I'm glad to see PCC get into hardware, expecially Vol. 3 No. 3. I want to see PCC get a BASIC compiler or interpreter going. I'm going to try and write one from the University of Illinois report. It'll be a long road. I like Tiny BASIC for a starter. Do you think it would be possible in 2–3K? Yup - next issue!

What is the address of the Rienold Publishing Co. with the book *Anatomy of a Compiler*? VAN NOSTRAND REINHOLD, 450 W. 33RD. ST., NY 10001

Good News! Bill Serviss in Dewitt, Mich. has started the "Mid-Michigan Micro Users Group." He found me through my last letter in PCC. There are about 12 people in the group. It is growing. Bill has a modified Mark-8 he built on perf board. He has a ¼k at the moment. Three or four more members plan to build Mark-8's.

As I told you in my last letter I was building a Mark-8. With Bill's tender care it did work for a while. Hopefully by the end of the summer I'll have enough dough scrapped together to have a 5K Mark-8, with TV typewriter, ASCII keyboard, cassette interface and BASIC or FORTRAN software.

Larry Miller
826 Halstead Blv.
Jackson, MI 49203

COMPUTER ART CONTEST

Two double winners were produced in the seventh annual computer art contest sponsored by the Kiewit Computation Center at Dartmouth College this spring.

At recent informal ceremonies, Math Prof. Thomas Kurtz, director of the computation center, awarded prizes to the following:

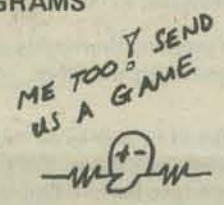
James Browning, a Dartmouth freshman from Hanover, who won the first prize of $75 with a dramatic audio-visual presentation entitled "WHE" and who tied for second prize and a $50 award with an untitled entry depicting an interacting flow of colors;

Robert B. Clyman, a sophomore from Great Neck, NY, who shared second place honors and also received a $50 award with a lovely geometric design entitled "BLOSSOMS";

Lynn Brooks, a junior from Grand Rapids, Mi, whose stylized representation called "PONIES" took third place and a $25 award and whose second entry entitled "OWLS" received honorable mention; and

Kevin U. Cohan, a freshman from Sydney, Australia, whose untitled work also received an honorable mention.

WANTED: GAME PROGRAMS



I'm interested in good, unique, and comprehensive game program – mainly my subject of interest are –

sports simulations (baseball, hockey etc.)
card game simulations
table game simulations (monopoly, etc.)
spacewar games

or any other interesting program. I'm a programmer who has taken game programming as my hobby. I have lots of spare time to work on program, so just send me listings since I don't mind typing in programs that interest me.

Thanks, and send them to:

Gary Trapp
310 Julian St.
Denver, Co. 80219

WANTED:8008 BASIC

I am currently working on an RGS-008A micro (8008 based) and already have the TVT I built with some mods (send them in when I get them working decent) and a cassette interface (RGS design).

I'm a senior at Purdue in EE with strong interest in CS. I'm also President of our ACM. As soon as I get them modified off our tape, I'll send you our complete game file and some of our better pictures.

Hurry with the BASIC interpreter for the 8008. Any problems I become aware of, I'll put on our University Computer Mail service, which includes undergrads, grads, profs, (all interested in software or hardware) along with all our system programmers.

Fred Rosenbaum
1-7 Ross Ade Dr.
West Lafayette, IN 47906

See page 12, this issue.

Technology, McDonald's Collide As Students Best Burger Bonanza

By Catherine Arnst
Of the CW Staff

PASADENA, Calif. – McDonald's Restaurants, whose hamburgers have taken their place along with Mom and apple pie as a piece of Americana, was recently confronted by a computer and 26 students from the California Institute of Technology (Cal Tech) following another American tradition – free enterprise.

It started when 187 McDonald's in five counties of southern California held a sweepstake during March. The $40,000 worth of prizes included a new sports car, a year's free groceries, a station wagon and free McDonald's coupons.

Entrants were required only to be a resident of one of the five counties and fill out either an entry blank or a three-by-five piece of paper with their name and address. No purchase was required and there was no limit to the number of times each person could enter.

The Cal Tech students, headed by senior John Denker, realized these rules presented them with an opportunity to turn their DP training to a money-making advantage.

The students used the school's IBM 370/158 to print out 1.2 million entry blanks with their names on them. Denker said enough paper was used to cover "two and one half football fields or [reach] higher than a three-story building."

The program they wrote consisted of four simple lines of Fortran. Although Denker admitted it probably would have been more practical to have a regular printer do the entry blanks, the students

(Continued on Page 4)

had ready access to the computer and it was faster.

On the final day of the contest the students went to 90 McDonald's in the specified counties and started stuffing the entry boxes. Their computerized entries made up over one-third of the 3.4 million total number of entries.

McDonald's Not Pleased

McDonald's was not delighted with the students' high level of participation in the sweepstakes. Although Denker claimed their entries are legally valid, Ron Lopaty, president of the McDonald's Operator's Association of Southern California, said he feels "the students acted in complete contradiction to the American standards of fair play and sportsmanship."

The contest's purpose, he said, was "to give customers an opportunity, in a time of economic stress, to win free groceries and transportation. So you can understand our displeasure when their chances of winning were greatly reduced by the Cal Tech students using an unfair advantage of computerized entry blanks."

Part of the public agreed with him in letters and phone calls to both McDonald's and Cal Tech. The state's attorney general even received a petition signed by over two dozen southern California residents which said "the use of equipment at a state or federally funded college, university or institution for the pursuit of personal interest, not to mention cheating American consumers, is an absolute outrage."

As for Cal Tech, it has taken no position on the issue, claiming it was the students' private endeavor.

Lopaty said McDonald's has agreed "to honor as 100% valid all the Cal Tech students' 1.2 million computerized entries" and, in fairness to the other entrants, will hold a second drawing in which all the computerized entries will be excluded and duplicate prizes of any won by the students will be awarded again.

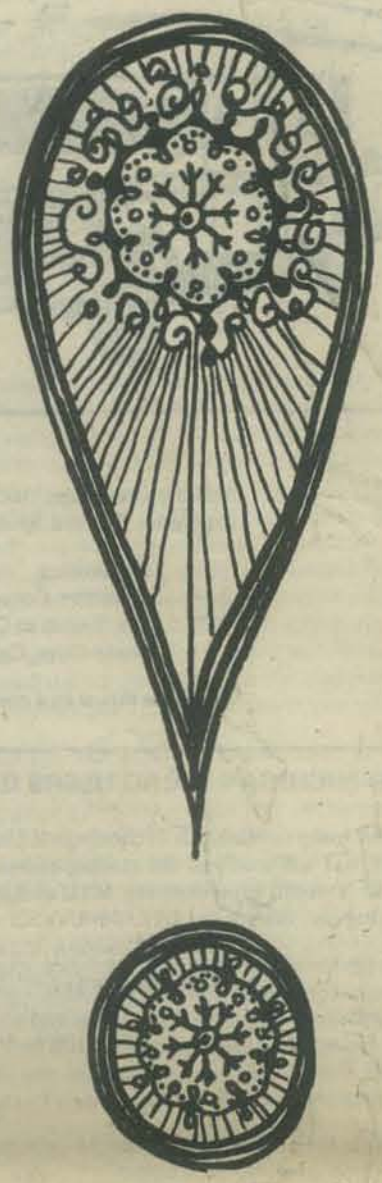
For the students, the McDonald's caper, as they call the affair, has paid off. They have already been notified they've won a Datsun 710 station wagon, a year's free supply of groceries and innumerable $5 gift certificates.

"Part of the loot will be used to finance improvements in Page House, our residence here at Cal Tech," Denker said. "The rest will be donated to charity."

Denker was dismayed at the restaurant chain's reaction to the incident, saying he doesn't feel they violated American standards of fair play.

"Just because it is unexpected doesn't mean it's unfair," he explained. "We feel that by accepting the challenge to enter as often as you wish, we have acted in accordance with the best ideals of American sportsmanship."

There are those who agree with him, and Cal Tech garnered a prize of its own from one of them. The Burger King chain of restaurants, McDonald's arch rival, has awarded $3,000 to the school to set up a "John Denker Scholarship" in honor of the student who masterminded the scheme.



MANTRA

The Prayer wheel programs I mentioned in an earlier letter will be released shortly – as soon as I get time to make 'em pretty. These are PDP's (Public Domain Programs) that were written in 360 assembler, so they should run on any 360, 370 or a Spectra machine.

Program 1 is named DISCMANT. The current version is device-dependent. It writes a single logical record that fills an IBM 2316 disk pack track (7294 8-bit bytes) with 405 catenated copies of the 18 character (EBCDIC) string –

'OM␣MANI␣PADME␣HUM'

To pad out the last 4 bytes, 'OMOM' is used.

After the string is written, the data set it comprises will stay on the disc volume until it is scratched. Meanwhile the disc pack is spinning away, whether or not the CPU is operating . . . cranking out over 10^9 PRE's per day. (A PRE is a 'prayer-revolution – equivalent is a measure of prayer-wheel performance. a PRE is defined to have the same effect as the physical rotation thru 2π radians, of a theologically acceptable graphic representation of the Tibetan Buddhist mantra

OM MANI PADME HUM.

(invented it myself, I did)).

It's a good idea to put this on a "public" or "system-resident" volume – a volume that's mounted all the time . . . evidently the thing is to keep a prayer wheel spinning all of the time.

Program No. 2 is named OM. Essentially it keeps the same mantra circulating in the four floating point registers (again, 360) in a tight loop – runs until time for job (or job step) is up . . . at which time the OS grabs control and bombs the program. This insures that the time the user requests is spent running the mantra around rather than in "cleaning things up", as "good programming practice" would dictate (closing data sets, returning control to OS, etc., etc.).

I don't think it's as useful as DISCMANT (for DISC MANTra) but who knows?

I'm not a Buddhist myself – the person I wrote the programs for is. I'm not exactly sure what a prayer wheel does – except that what ever it is, these programs seem to do it – or so I have been assured by the person who asked me to write them.

You'll get one source deck plus "assembly, link-edit and go" listing per program.

If anybody wants, I'll supply source or object modules (card or 9 track tape, Std label, even parity) for cost of job plus materials and postage.

Why haven't the Tibetan Buddhist monasteries picked up on this? I'd be super cheap – it's just a dense storage medium and all you have to do is spin it – no I/O required, so (Look, Ma!) no beads . . .

Both program thoroughly documented – not all computer jargon, either.

I'd like to be able to do "socially redeming" things like that for a living. I'm getting to the point where I'm going to have to find work . . . I'm taking COBOL –a draggy language– and I'm totally turned off to the business world. I'd like to find a job doing meaningful user-service-oriented (or systems, utilities, networking, graphics) computer work. If anybody out there is looking for a good software ex-hardware (but not turned off to same) person, please contact me. I know the following languages in and out –

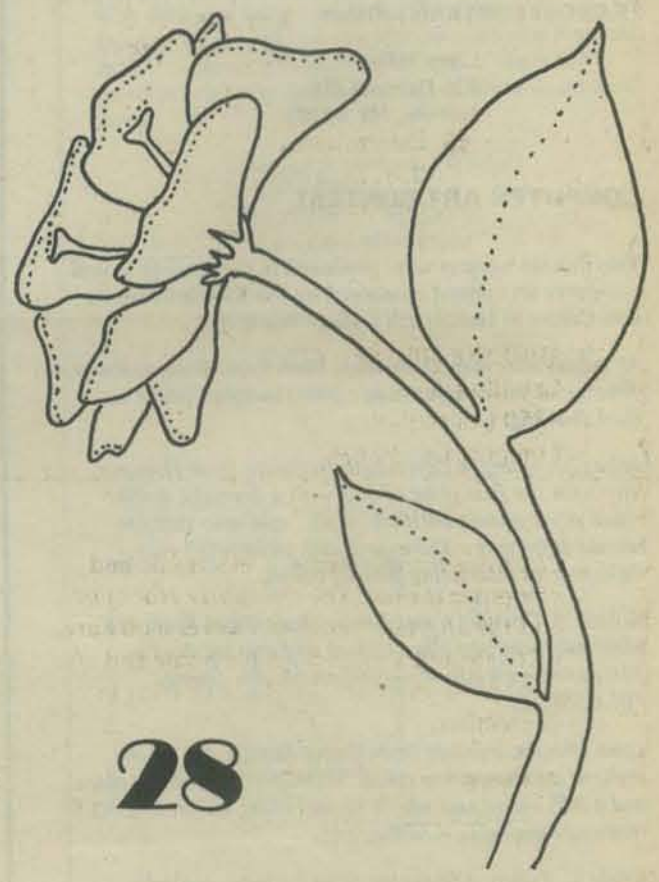
360 Assembler + macro langauge – written some nice system and user utilities, and OS/MVT system mods.
FORTRAN IV + Calcomp plotter – wrote a batch LIFE program, lots of options.

Less well, but have written and debugged programs in –

MIXAL	NASIC	BASIC
Turing machines	360 JCL + PROCS	
COBOL	LISP	
PL/I	APL	

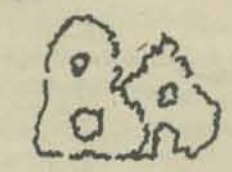
I think I'm creative and resourceful enough to do such a job (but I don't feel creative and resourceful enough to find one – not without feedback enyway) . . . and hopefully contribute to the demystification of computers.

Kurt Cockrum
"People's Free Bit Crunching Facility"
3398 Utah
Riverside, Ca. 92507



Dragon Droppings

Hi mortals. Here are some of the things I peruse each month for information, inspiration and entertainment.



NEWS OF THE TRADE

Magazines, newspapers, etc. Some free (to "qualified individuals"), some expensive some in between. Write for info.

* COMPUTER DECISIONS
  Hayden Publishing Company
  50 Essex St.
  Rochelle Park, NJ 07662

* COMPUTERWORLD
  797 Washington St.
  Newton, MA 02160

* COMPUTERS AND PEOPLE
  (Formerly Computers and Automation)
  Berkeley Enterprises
  815 Washington St.
  Newtonville, MA 02160

* DATAMATION
  Technical Publishing Co.
  35 Mason Street
  Greenwich, CT 06830

* INFOSYSTEMS
  Hitchcock Publishing Co.
  Hitchcock Bldg.
  Wheaton, IL 60187

* MODERN DATA
  P. O. Box 369
  Hudson, MA. 01749

* MICROCOMPUTER DIGEST
  2368–C Walsh Ave.
  Santa Clara, Ca. 95050

* MINICOMPUTER NEWS
  Benwill Publishing Corp.
  167 Corey Road
  Brookline, MA 02146

HOBBY STUFF

If you are building your own, you probably will like these –

* Computer Hobbyist
  Box 295
  Cary, NC 27511

* Electronotes
  B.A. Hutchins
  60 Sheraton Drive
  Ithaca, NY 14850

* MICRO-8 NEWSLETTER
  Cabrillo HS
  4350 Constellation
  Lompoc, Ca. 93436

Electronotes specializes in electronic and computer music. The *Computer Hobbyist* and the *Micro-8 Newsletter* cover hardware, software and applications for home and personal computers. And . . . starting in September,

* BYTE
  Peterborough, NH 03458

INSPIRATION

If you like games or simulations or math recreations or . . . try the following

Flying Buffalo's Favorite Magazine
P. O. Box 1467
Scottsdale, AZ. 85252

Journal of Recreational Mathematics
Baywood Publishing Company
43 Central Drive
Farmingdale, NY 11735

Simulation/Gaming/News
Box 3039, University Station
Moscow, ID. 83843

STAR TREK, ANYONE?

Get your official STAR TREK catalog from

Lincoln Enterprises
P. O. Box 69470
Los Angeles, Ca. 90060

Maybe you will want to join the STAR TREK Animated Fan Club.

FROM YOUR FRIENDLY COMPUTER VENDOR

Newsletters for your enjoyment and enlightenment.

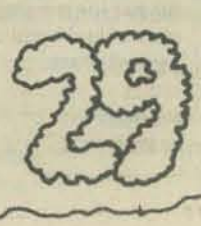
* Computer Notes
  MITS, Inc.
  P. O. Box 8636
  Albuquerque, NM 08708

* EDU
  Digital Equipment Corp.
  Maynard, MA 01754

* HP Educational Newsletter
  Hewlett Packard
  11000 Wolfe Road
  Cupertino, Ca. 95014

Least Expensive BASIC?

Until September 15, 1975, you can buy an 8K ALTAIR 8800 kit, complete with BASIC, for $995. Add the terminal of your choice and start computing. For more info, write MITS, 6328 Linn NE, Albuquerque, NM 87108



Book Reviews

how two ex-students turned on to pure mathematics and found total happiness

SURREAL NUMBERS

a mathematical novelette by D. E. Knuth

What's half of infinity? What is less than 2 but neither <0, =0, nor > 0? Answers to these and other mystifyiers are revealed in **Surreal Numbers**, a mathematical novel-in-dialogue by Don Knuth of Stanford's Computer Science Department.

The story tells of Bill and Alice, dropped out and isolated by choice on a remote beach by the Indian Ocean. In a moment of boredom, they uncover an ancient stone revealing the fundaments of a new number system designed by J. H. Conway of Cambridge University. Together, they turn on to evolving a theory of Conway's numbers, and the reader follows their adventures and frustrations as they start with nothing and together create a variety of appealing notations, the entire set of real numbers ... and more.

Familiarity with algebra, set notation, induction and deduction are the only prerequisites to appreciation of this entertaining story. It is best read quickly in one or two sittings, with out dwelling on the detailed steps of proofs. The exciting discoveries are at the end, and enough can be picked up in a quick reading to understand them. If you later find your thoughts filled with Conway numbers, many of Alice and Bill's discoveries cry out for formal proofs, and if further relief is needed, there are 22 explicit excercises in the back. From there, you can carve out your own branch of the new theory.

Knuth -- an energetic mathematician, computer scientist, teacher, writer, organist and sage -- likes to turn people on to elegant thinking, and in this book goads the educational establishment to follow suit. His closing words might well be stamped on the walls of every math, science, and engineering classroom in our universities:

In my opinion, the two weaknesses in our present mathematics education are the lack of training in creative thinking and the lack of practice in technical writing. I hope that the use of this book can help make up for both of these deficiencies.

It's a good start.

Larry Tesler
Menlo Park, Ca.

Surreal Numbers by D. E. Knuth
1974, 119 pgs., **$3.95.**

Travels in Computer Land by Ben Ross Schneider, Jr.
1974, 224 pgs., **$5.95**

These two books are available from:
Addison - Wesley Publishing Co., Inc.
Reading, Ma 01867

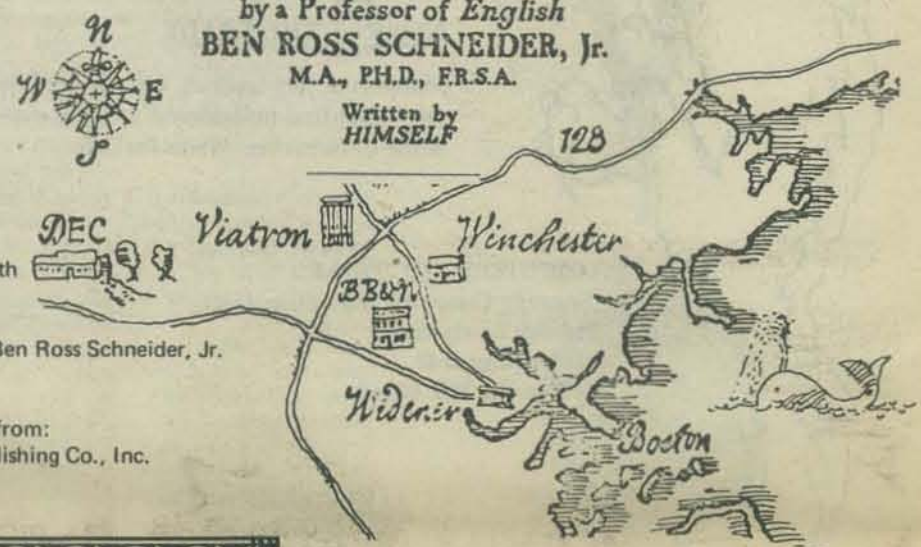
TRAVELS IN COMPUTERLAND;

OR,
INCOMPATIBILITIES *and* INTERFACES

A *Full and True* Account of the IMPLEMENTATION of the *LONDON STAGE* INFORMATION BANK

by a Professor of *English*
BEN ROSS SCHNEIDER, Jr.
M.A., PH.D., F.R.S.A.

Written by
HIMSELF



This book describes the experiences of a university professor who embarks on a computer project. He starts as an innocent and ends pretty much the same, though with his mind and vocabulary somewhat broadened by the experience. The book is not a textbook, nor notably informative about computers, though there are undoubtedly unfortunates for whom it would have been extremely valuable. Its instructional merit will be appreciated by those with comparatively ambitious projects and large systems, but its real attraction is as literature.

The author writes from the detached viewpoint of a scholarly observer. His reflective and introspective style belongs to a long vanished, more leisured age, but the gentle humor with which he regards both himself and his surroundings does not disguise the almost crystalloid precision of a highly trained mind. His perceptive observations sometimes flash with insights well worth consideration, especially by those whose understanding of computers and computer people is extensive.

In short, the book is a two way window. Those with no knowledge of computers may peer through it at a strange landscape -- computer people may find through it much that is strange in farmiliar surroundings. One warning, though, the reviewer is an Englishman.

Keith Britton
Loma Mar, Ca.

TEACH YOURSELF
basic

volumes 1&2

64 pgs. ea., $1.95 ea.

Tecnica
EDUCATION CORPORATION

now available at:
PCC Bookstore

One of the first BASIC primers to receive wide acceptance in the junior high and high school market has been revived by PCC bookstore. **Teach Yourself Basic I & II** was written in 1970 by your friendly dragon for Tecnica Education Corporation (remember them?). Written at about an seventh grade math level, it received immediate acceptance for use in grades 7 - 14. It nearly disappeared when Tecnica moved to Utah, but we just found it again and offer it to you. Book I covers teletype fundamentals and BASIC instructions: LET, PRINT, GO TO, FOR NEXT, INPUT, and a few other fundamentals in that popular "read it and try it" style. Book II deals with IF THEN, ARRAYS, and other seemingly more sophisticated instructions for the novice. When you buy it, you may recognize that **Teach Yourself Basic I & II** have been reproduced in their entirety by DEC in EDU Handbook. This form of the book is an alternate way to buy this product. The price is right, the quality there. Try it... lots of others have liked it.

Leroy Finkel
Menlo Park, Ca.

Teach Yourself BASIC I & II

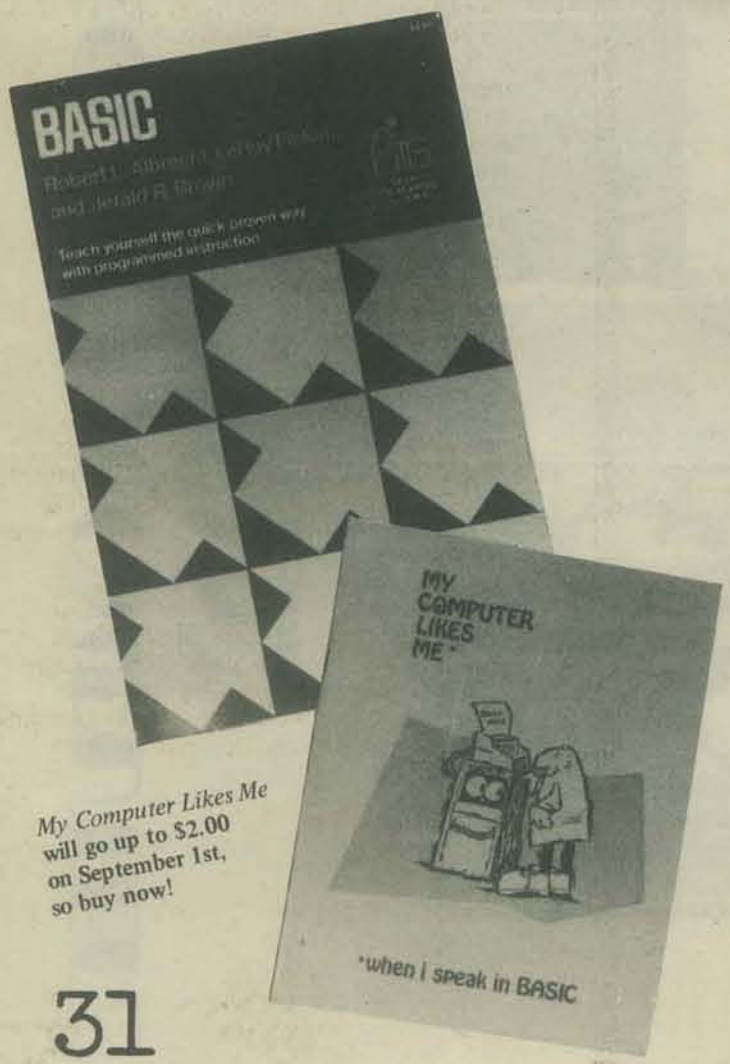
Bob Albrecht, 1970, p. 64 – $1.95 each
Published by TECNICA

Fun And Games With The Computer

Edwin R. Sage, 1975, p. 351 – $5.95
Published by ENTELEK

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 & 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, Bob Albrecht, 1972, p. 64 – $1.49
101 BASIC GAMES, Ed. David Ahl, 1974, p. 250, – $7.50
PROBLEMS FOR COMPUTER SOLUTION, Gruenberger & Jaffray, 1965 – $7.25
PROFESSOR GOOGOL, Sam Valenza, Jr., 1973, p. 144 – $3.25
PROBABILITY, D. J. Koosis, 1973, p. 163 – $2.95
PCC GAMES, Program Listings – $2.00
SELLING WHAT YOU MAKE, Jane Wood, 1973, p. 111 – $2.25
STATISTICS, D. J. Joosis, 1972, p. 282 – $3.95
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. 157 – $6.95
DIGITAL LOGIC CIRCUITS, Sol Libes, 1975, p. 184 – $5.98

A PRACTICAL, LOW-COST HOME/SCHOOL MICROPROCESSOR SYSTEM
reprint from COMPUTER magazine, 1974 – $1.00



My Computer Likes Me will go up to $2.00 on September 1st, so buy now!

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	

Mail order to:

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

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

PCC

editor BOB ALBRECHT
production MARY JO ALBRECHT
circulation LOIS BRITTON
contributors

Keith Britton
Kent Cross
Lee Felsenstein
LeRoy Finkel
Gordon French
Bob Mullen
Candy Nadel
Dennis Allison
Robin Allison

CONTENTS

PCC is a not-for-profit corporation. The newspaper is about recrational and educational uses of computers – computers for everyone. PCC is published 6 times during the year.

Subscriptions are $5.00 for 6 issues. ($6.00 outside the U.S.A. – surface mail; $12.00 air mail.) Subscriptions begin with the July issue.

Next Issue Games, Games, Games! Some in BASIC, some in nitty gritty machine language for small home computers. And (we hope) . . . the first in a series on voice input to your personal computer. And - a bunch of pages on TINY BASIC for 8080!

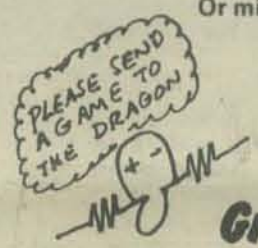
Back issues get them while they last at the following low low prices

Vol. I, Nos. 1 – 3† $1.00
Vol. II, Nos. 1, 3, 4, 5, 6† $4.00
Vol. III, Nos. 1 – 5 $4.00

† Sorry, we are out of Vol. 1, Nos. 4 & 5 and Vol. II, No. 2

Or mix up individual issues –

2 – 9	$0.80 each
10 – 99	$0.70 each
100 +	$0.60 each



Group subscriptions (all mailed to the same address)

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

TO ORDER send check or money order to:

PEOPLE'S COMPUTER COMPANY

P.O. Box 310 Menlo Park, Ca. 94025

Name _____
Address _____
_____
zip

○ **regular** 1 year $5 ○ 2 years $9 ○
○ **group** (how many?) _____

○ **back issues**
○ vol. I $1
○ vol. II $4
○ vol. III $4

Printed by Nowels Publications 640 Roble Ave., Menlo Park, Ca. 94025