



# Homebrewery vs the Software Priesthood

**Mike Wilber**  
920 Dennis Dr  
Palo Alto CA 94303

**David Fylstra**  
PO Box 10051  
Stanford CA 94305

The movement of computers into people's homes makes it important for us personal systems users to focus our efforts towards having computers do what we want them to do rather than what someone else has blessed for us.

The movement towards personalized and individualized computing is an important threat to the aura of mystery that has surrounded the computer for its entire history. Until now, computers were understood by only a select few who were revered almost as befitted the status of priesthood. The arts of designing and programming computers have long been regarded as sacred knowledge beyond the reach of the nonspecialist. Indeed, the journeymen of the trade have protected their privileged position by keeping their knowledge to themselves. These high priests and acolytes of the holy alliance of logicians (HAL) have dominated the field so far.

The movement of computers into people's homes makes it important for us personal systems users to focus our efforts toward having computers do what we want them to do rather than what someone else has blessed for us. If personal computing users freely share their hard-won information and even their programs, then this community of users can become quite adept at bending their computers to their own needs and wants.

When computers move into people's homes, it would be most unfortunate if they were merely black boxes whose internal workings remained the exclusive province of the priests. It is fine to use them as black boxes as long as they do what you want, but

the computer's owner should be able to modify its behavior to suit merely personal preferences. That is, computer literacy should be widespread. Now it is not necessary that everybody be a programmer, but the potential should be there. This understanding must be removed from the private reserve of the select few just because computers will be in the hands of the many.

Freely available software is perhaps the key to the independence of the homebrewer.

There are several indications that personal computing users are likely to be highly motivated to generate freely available software. One can be found in the amateur radio community: Hams have a long tradition of freely sharing their feats of engineering. In fact, it is quite natural to be proud of one's accomplishments, and that pride finds quite natural expression in telling all the details to anybody showing the slightest signs of interest. Already there are some signs that computer hobbyists have similar pride in their software achievements. Dr Dobb's *Journal* contains several examples of people contributing adaptations and extensions to Tiny BASIC. The mode of software development is likely to follow the example of program development among the world of paid programmers. For a while, people will contribute incremental improvements to a program (or a concept). During this first half of the cycle, the program becomes progressively more powerful and progressively messier until it is quite hard to understand or modify. Finally, somebody gives up in disgust and rewrites the program in the marvelously clear way that had been growing increasingly conspicuous by its absence. Then another cycle starts, but its starting place is far more advanced than the previous cycle had reached.

"...these are, and of right ought to be, Free and Independent..."

— John Hancock, et al

Personal computing people stand to be largely independent of the priesthood because they are strikingly sophisticated and because they freely share their ideas. A very good example of both these traits can be found in the nearly spontaneous generation of Tiny BASIC through the medium of the People's Computer Company and Dr Dobb's *Journal of Computer Calisthenics and Orthodontia*. One issue published some rough design notes for a machine independent Tiny BASIC, but that was only the beginning. The next few issues published refinements on the design and later ones included an implementation in an interpretive language and then both octal and annotated source programs realizing the interpreter and the entire system in 3 K of 8080 code. To top it off, the whole project was done by far-flung individuals in less than a year.

While Tiny BASIC is a very striking example of what amateurs can do when they work together, we cannot afford to ignore its extreme dependence on good fortune to bring it to pass. Your own copy of BYTE magazine is another example; it is the result of one man's frustration at making his own computer work and his desire to let others profit by his experience. We've been very lucky to have a few people with high ideals to point the way for us, but we would be ill advised to depend on having these fortunate circumstances continue. The time is ripe for the community of personal computing enthusiasts to start thinking seriously about supplying its own steam to back up the energies put out by a few people with strong motivations to help launch the personal computing movement. It's launched now, and we have to provide the impetus and direction to make sure it develops in a way beneficial to the community at large.

A good example of a means to distribute software which divides the effort fairly and in a way nobody seems to mind is the software exchange of the Homebrew Computer Club in the San Francisco Bay Area. At each meeting (every two weeks) there is a table covered with paper tapes of programs contributed by all and sundry. Anybody is welcome to take any tape at all, subject only to the proviso that each copy taken from one meeting be replaced by at least one



## UNBELIEVABLE!!!!

### The Intecolor® 8001 Kit

A Complete 8 COLOR Intelligent CRT Terminal Kit

**\$1,395**

#### "Complete" Means

- 8080 CPU • 25 Line x 80 Character/Line • 4Kx8 RAM / PROM Software
- Sockets for UV Erasable PROM • 19" Shadow Mask Color CR Tube
- RS232 I/O • Sockets for 64 Special Graphics • Selectable Baud Rates to 9600 Baud • Single Package • 8 Color Monitor • ASCII Set
- Keyboard • Bell • Manual

And you also get the Intecolor® 8001 9 Sector Convergence System for ease of set up (3-5 minutes) and stability.

#### Additional Options Available:

- Roll • Additional RAM to 32K • 48 Line x 80 Characters/Line • Light Pen
- Limited Graphics Mode • Background Color • Special Graphics Characters
- Games

#### ISC WILL MAKE A BELIEVER OUT OF YOU.

PE 276

Send me \_\_\_\_\_ (no.) Intecolor® 8001 kits at \$1,395 plus \$15.00 shipping charges each.

Enclosed is my  cashier's check,  money order,  personal check\*

\$350 deposit/kit for C.O.D. shipment for \$ \_\_\_\_\_

NAME \_\_\_\_\_

ADDRESS \_\_\_\_\_

CITY \_\_\_\_\_ STATE \_\_\_\_\_ ZIP \_\_\_\_\_



**Intelligent Systems Corp.** 4376 Ridge Gate Drive, Duluth, Georgia 30136  
Telephone (404) 449-5961

\*Allow 8 weeks clearance on personal checks.  
Delivery 30-60 days ARO



Software exchange is a two way street: He or she who uses an application or system's program from a community library is assumed to be willing and able to provide programs of equivalent value for others to use.

copy at the next meeting. A few cautions, however, must be repeated every few meetings: that people label the tapes they bring back, that they take reasonable steps to ensure the accuracy of their copies and that they only contribute software with the author's consent. Homebrewers have good intentions but still need occasional reminders to keep them from getting careless.

It is most unfortunate that some people give free distribution to software against the author's wishes. In fact it's usually illegal, and anyone caught could face a heavy fine because the crime is new enough that many think prominent examples should be offered to reduce its frequency. Until that lucky day, the main people being deterred are the people who have contributed the software that has immeasurably helped the whole hobbyist movement get started. Even though we have a tremendous potential for generating our own software, we still owe a tremendous debt (of gratitude besides the money) to those who have brought us Altair BASIC and 6800 BASIC, and who may have enough faith in us to bring us APL and some truly groovy text editors. Freely exchanged software should be truly free and untainted by ripoffs or by the appearance of ripoffs.

The extreme ease of software theft could present a real barrier to free interchange of good software because many valuable people could understandably be reluctant to become very deeply involved in a forum where such ripoffs are commonplace. Theft is so much easier with software than with hard-

ware because software has two distinctive properties:

- A buyer cannot evaluate its benefit without extensive testing and use.
- The elementary operation in moving software is not to transport one copy but to generate another copy.

Another danger threatens free distribution of free software, and for some of the same reasons, a scarcity of documentation which is all too common in software. The temptation is strong: Somebody developing it understands it well and is concentrating on getting it to work at all and usually prefers not to be distracted by efforts to make it comprehensible to others. On the other hand, the task of documentation can easily take as much effort as the development itself. However, documentation is crucial to the value of a piece of software. Undocumented software is very hard to use and even harder to modify. We have no good solutions to either of these problems. The only thing to do is to repeatedly urge people to be mindful of the problems and to broaden their perspectives beyond the gains of the short term.

#### Telecommunications and the Community Information Exchange

Enthusiasts in this field can share software right now by banding together into clubs, but that medium limits sharing to small groups of people who live near one another. With telecommunications, people can share their programs with others living at long distances from one another. One vital ingredient to such remote communication is the ordinary telephone, which works wonders at spanning long distances between people. Telephones can have the same benefit for computers, if they are equipped with modems, which handle the translation between a computer's digital signals and the audio signals the telephone can handle. Then, one person can call another, and they can use that same call to connect their computers; one computer can run a special program to copy data from a cassette or memory to the phone line; and the other can run another special program to copy the information from the phone line onto its own cassette. Of course, the data being copied will likely be some program the two parties wish to share; and presto, you have an instance of software sharing at a long distance.

While that kind of person to person exchange is quite effective, it leaves room for improvement in several respects. It requires very close coordination between the

two parties, and it requires one phone call for each interchange of data. However, there is another possible mechanism, which can permit widely scattered users to communicate far more freely and with much looser coordination while improving on the economy of phone line usage. That would be a sort of "Community Information Exchange," a computer that would be continually prepared to automatically answer the telephone and would expect a computer to be placing the call. It would be located where a number of individuals (for example members of a local club) could reach it with a local call; it would provide bulk storage facilities, and it would accept commands in a very concise, well understood format from the computer which had called it. Then one subscriber could leave a program in the bulk storage and invite all other subscribers to that Community Information Exchange to copy it to their own systems at their leisure.

This is a very powerful means of broadcasting software among a local community, but it has implicit in it a means of broadcasting opinions and news too. It requires a means of transmitting plain English text between people just so the people will know which programs they can or might wish to communicate to their computers. Once the

individual subscribers can communicate words among themselves, they can communicate much more than news about the latest programs available and how to use them. They can also tell one another about the problems they are having with some program they recently picked up or even about problems they are having with some hardware they recently bought. The computer community can find a great deal of strength in freely sharing that kind of information, in addition to sharing their programs. Free communication of information of all types can greatly enhance the community's resistance to inferior products, and acceptance of superior products.

The Community Information Exchange is not limited to the local communication described thus far. In the dead of night, when telephone traffic is reduced and the transcontinental rates are low, a CIE in one locality can call a CIE in another locality. Then they can send programs and other data back and forth. Of course, they have to know just what should be sent where, and they could be told by their subscribers. The commands they will accept from their subscribers could direct them to copy a file to or from some remote CIE. Nor is it necessary for a CIE to directly call another to

Imagine a Community Information Exchange, complete with telecommunications access ports, mass storage and an accounting algorithm to keep track of operating expenses attributed to each user's activities.

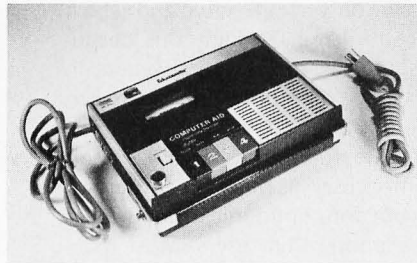
**NOTE:** The term "Community Information Exchange" was inspired by Michael Rossman.

#### MODEL CC-7 SPECIFICATIONS:

- A. Recording Mode: Tape saturation binary. This is not an FSK or Home type recorder. **No voice capability. No Modem. (NRZ)**
- B. Two channels (1) Clock, (2) Data. OR, Two data channels providing four (4) tracks on the cassette. Can also be used for Bi-Phase, Manchester codes etc.
- C. Inputs: Two (2). Will accept TTY, TTL or RS 232 digital.
- D. Outputs: Two (2). Board changeable from RS 232 to TTY or TTL digital.
- E. Runs at 2400 baud or less. Synchronous or Asynchronous. **Runs at 4800 baud or less. Synchronous or Asynchronous. Runs at 3.1"/sec. Speed regulation  $\pm .5\%$**
- F. Compatibility: Will interface any computer or terminal with a serial I/O. (Altair, Sphere, M6800, PDP8, LSI 11, IMSAI, etc.)
- G. Other Data: (110-220 V), (50-60 Hz); 3 Watts total; UL listed 955D; three wire line cord; on/off switch; audio, meter and light operation monitors. Remote control of motor optional. Four foot, seven conductor remoting cable provided. Uses high grade audio cassettes.
- H. Warrantee: 90 days. All units tested at 300 and 2400 baud before shipment. Test cassette with 8080 software program included. This cassette was recorded and played back during quality control.

ALSO AVAILABLE: MODEL CC-7A with variable speed motor. Uses electronic speed control at 4"/sec. or less. **Regulation  $\pm .2\%$**   
Runs at 4800 baud Synchronous or Asynchronous without external circuitry.  
Recommended for quantity users who exchange tapes. Comes with speed adjusting tape to set exact speed.

#### DIGITAL DATA RECORDER \$149.95 FOR COMPUTER or TELETYPE USE Any baud rate up to 4800



Uses the industry standard tape saturation method to beat all FSK systems ten to one. No modems or FSK decoders required. Loads 8K of memory in 17 seconds. This recorder, using high grade audio cassettes, enables you to back up your computer by loading and dumping programs and data fast as you go, thus enabling you to get by with less memory. Can be software controlled.

**Model CC7 . . . \$149.95**  
**Model CC7A . . . \$169.95**

**NATIONAL multiplex**  
CORPORATION

**NEW — 8080 I/O BOARD with ROM.** Permanent Relief from "Bootstrap Chafing"

This is our new "turnkey" board. Turn on your Altair or Imsai and go (No Bootstrapping). Controls one terminal (CRT or TTY) and one or two cassettes with all programs in ROM. Enables you to turn on and just type in what you want done. Loads, Dumps, Examines, Modifies from the keyboard in Hex. Loads Octal. For the cassettes, it is a fully software controlled Load and Dump at the touch of a key. Even loads MITS Basic. Ends "Bootstrap Chafe" forever. Uses 512 bytes of ROM, one UART for the terminal and one USART for the Cassettes. Our orders are backing up on this one. No. 2SIO (R)

**Kit form \$140. — Fully assembled and tested \$170.00**

Send Two Dollars for Cassette Operating and Maintenance Manual with Schematics and Software control data for 8080 and 6800. Includes Manual on I/O board above. Postpaid.

Master Charge & BankAmericard accepted. On orders for Recorders and Kits please add \$2.00 for Shipping & Handling. (N.J. Residents add 5% Sales Tax)

3474 Rand Avenue, Box 288  
South Plainfield, New Jersey 07080  
(201) 561-3600



have access to its file storage. Instead, commands and data could be relayed from one CIE to another until they finally reach their destination. Then, many isolated CIEs would behave like a vast network capable of transmitting software across the country overnight.

We are describing a communication network which can be very effective and which is highly decentralized. In fact, this decentralization is crucial to its effectiveness in promoting free communication between individual computer people. For example, computer manufacturers now organize their customers into user groups in order to provide a forum for communication of ideas and programs among their users. Most communication is channeled through publications controlled by the manufacturer, however; and this all too often results in the encouragement of software and viewpoints which are consistent with those of the manufacturer. On the other hand, it is not necessary to use a centralized forum to encourage people who need it: most of that interaction is on a person to person basis in almost any group.

All well and good, you might say, but just how does the personal computing user compare with the journeyman programmer? Well, the computer amateurs live in a world in which wholesale copying of programs is nearly inevitable. Actually, that represents a

healthy trend in two ways: the person using the copy benefits by its availability; and the program's originator benefits by having helped spread a good example of the programmer's art. Of course, such freely exchanged software may well be worthless at best. That does seem a minor penalty, though, for all the advantages that stand to be gained by ready availability. That is one example of the difference between the two kinds of programmers. That is, a functioning program is required of the professional, while it is merely desired by the amateur user who figures he or she will have to patch and customize anyway. Someone paid to write a program has to make it do what the client wants; a personal computing user has only personal preferences to satisfy. Somebody paying for a program has a right to expect that it will be reasonably efficient in order to conserve the money spent on computer time, while your patience is a much stronger constraint for you than the cost of your computer's time. Somebody paying for a program is quite likely to need the documentation oriented toward people unskilled in the technicalities, while fellow enthusiasts won't need to have all the details explained to them. Finally, deadlines are quite firm in the commercial world but of considerably less importance among amateurs [*except for those who fill magazines once a month! . . . CH*].

While the priests who market the old time software religion can help the personal systems user, such users should be mindful of the benefits to be gained from a healthy measure of independence. If you can get your hands on the symbolic form of freely exchanged software, you can revise it to suit your own needs. That's considerably easier than trying to convince somebody else who supplies a high priced package to make your favorite changes, especially since the other person may differ from you in values, priorities and notions of demand. For the traditionalist software source, a request from a single person could easily seem to represent too narrow an interest to motivate a change or patch. If program source listings are distributed freely and nearly every user has the requisite skills to make patches, then the person wanting something changed will probably be the person making the change. Of course, the end result is nearly the best of all possible worlds: Your home computer will do what you want it to do, and it will do it in the way you want it done. It will not do what somebody else decided it was reasonable for you to want it to do and in a way that it was convenient for somebody else to have it done. ■