The Small Computer Magazine

kilobaud

Understandable for beginners . . . interesting for experts

May 1978 / Issue #17 / $2.00 / DM 7,50 / Sfr 8,10 / Ffr 16,0 / UK £2

5.00

tomorrow's computer here today . . .

the bytemaster

only from the Digital Group

Digital tape based starts at $1995 kit.
Delivery within 60 days.

For details write or call today.

the digital group

P.O. Box 6528 Denver, CO 80206 (303) 777-7133

PUBLISHER'S REMARKS

Wayne Green

The Digital Group

As a continuing part of my plan to personally visit as many of the major firms in the microcomputing industry as possible, I recently stopped by The Digital Group facility in Denver. They've recently moved to new and larger quarters, a move I envy. Success is almost as difficult to cope with as failure—it's just a lot less painful. TDG has been going through growth problems—and they are many—fortunately, with some success.

One of the reasons for some of the horror stories about early Mits service was the over-success of the 8800 Altair. It is almost impossible to go through catastrophic growth without experiencing severe traumas. The "science" of selecting personnel is not yet perfect. It takes weeks or months to find out whether the peg is round or square, so the end result is that you have to train two or three people for each job—an expensive and time-consuming situation—and all this is happening under great pressure from frustrated customers.

You can imagine the things that can go wrong as you try to expand from a very small group —I think Mits had about 15 people when they announced the Altair—to ten and 20 times your original number. Everything that can go wrong will . . . repeatedly.

TDG has been going through this catastrophic growth and seems to be emerging reasonably well. Despite the pressures on them to deliver systems already being advertised, they've been doing their homework and have new systems ready for showing. That's even more remarkable when you consider that they've just moved to a new building.

The new building should hold them for a while. We're so packed in here at *Kilobaud*, with about 70 people jammed into an old 40-room house, that those modern, spacious offices—some not even being used yet—caused some slight pangs of jealousy.

The snow required by nearby ski areas tends to obscure buildings in New Hampshire. Here's Infotecs' plant, somewhere under the snow.

The Infotecs Bombshell

While most of the manufacturers in the microcomputer field have been concentrating on designing and selling hardware, with just enough software to get the hardware to sell, at least one firm has been taking a different tack.

Three years ago I looked over the amount of money *73 Magazine* was spending on outside data processing and was amazed to find it was around $2000 a month. For that amount of money we could have an in-house system that could do what was needed . . . plus much more. I sent out word to the computer world that I needed a computer.

The salesmen flocked in, each with wondrous descriptions of how great his system would be and with conspiratorial references to competing systems and their virtually fatal flaws. The main problem for me was that I couldn't understand much of what the salesmen were saying. They spoke computerese and apparently had not even a vestigial recollection of English.

In my efforts to learn computerese, I stumbled through computer hobby newsletters and microcomputers. I fell for microcomputers, hook, Mits line and sinker. The newsletters were OK, but there really should have been a magazine . . . so I started calling the editors of the computer-club newsletters to see if I could find someone who might be willing to take a gamble. Hal Chamberlin wasn't interested. Neither was Hal Singer, nor Bob Albrecht. I finally got down to a chap named Helmers who had a newsletter with 300 subscribers and he said, "OK, let's try it." On the same day, as I recall, I came up with the name for the new magazine: *Byte*.

During the five weeks after the decisions to get cracking on *Byte*, I rounded up authors and mailing lists from manufacturers, wrote subscription letters, wrote to all of the clubs, put out newsletters, etc. It was during this time that Cal Holt stopped by to try and interest me in buying a PDP8/A to take care of the *73/Byte* subscription list, bookkeeping, orders, etc. While Cal was trying to sell me on buying a PDP8/A with his programming, I was busy selling him on what was happening in microcomputers and how I envisioned the future of this new industry.

It took us just five weeks to get the first issue of *Byte* on the press. This was five weeks of day-and-night work for me, but the first issue ran 15,000 copies instead of my original hope of 2000. The magazine got off to a good start . . . and so did the microcomputer industry. In November I managed to lose the magazine . . . a story I hope the lawyers will eventually allow me to publish. There are too many lawsuits going now, so you'll have to wait that one out.

Now, flash ahead about 2½ years to January 1978. Cal Holt started calling and leaving messages for me to get up to see his plant in Manchester (NH). What with CES in Vegas and a ham industry conference in Aspen, plus a long siege of the flu in January, it took me a while. When I finally made it to Infotecs it was worth the trip.

Though I haven't been making a big deal out of it in *Kilobaud*, I have been discussing the change in the microcomputer market in our *Kilobaud* Newsletter, which goes to the industry. The fact is that hobby growth has essentially stopped, and most of the growth in the industry has been in sales to small business. Since I had predicted this right from the beginning, this has not come as any surprise to me.

Being as trapped by the computer hobby as anyone, I have no intention of putting it down. But my rational has been this: The

The CRT and CPU units are tested here as they are completed.

Here the complete systems are checked out and run in. The whole Infotecs microcomputer system is thus put together and checked out in a relatively small plant in Manchester, New Hampshire. How big a plant will they need a year from now?

The CRT, printer and disk that make up the Infotecs complete system. The 6100 CPU and memory are in the disk unit.

computer hobby is a very demanding one, requiring a lot of work and expense for the hobbyist who is actually going to understand computers and work with them. It is, in computerese, a nontrivial hobby.

I felt that this demand on the individual would be a limiting factor. How many people could we find who would take the enormous amount of effort required to become serious hobbyists? This had to be a limiting factor, whether it came at 100,000 hobbyists or 1,000,000. My predictions were more in the 100,000 range.

Why did I put the figure that low? Well, I know how easy it is to get a ham license as compared to understanding computers, and I know how the number of hams has been limited by the effort required to learn the theory and the code, both trivial compared to computers. It seemed like a reasonable yardstick.

Talks with people at computer stores during the last year have convinced me that my predictions were not far off. Most stores are reporting that sales to hobbyists have not changed seriously during the last year, but that sales to business have come along from nothing to about four or five times those to hobbyists. Those stores that are particularly hobby oriented have been reporting about equal sales between the two factions. You can quickly spot a hobby store when you walk in . . . you're ignored unless you are a hobbyist. I don't care if you have $10,000 burning a hole in your pocket, they will fawn all over a kid playing Star Trek and pointedly be deaf to any questions you may have.

One of the major problems stores face in selling systems to small businesses is the lack of business programs. Few businessmen want to spend $12,000 or so getting the hardware, only to have to sit down and write their own programs. Some stores have been busy writing programs in order to facilitate sales, but this is awfully expensive.

Imagine my surprise and delight to visit Infotecs and find that they had developed a complete microcomputer system of their own, including about the most comprehensive fuel-oil-dealer package I could imagine. The system is based on the Intersil 6100 chip, which emulates the PDP8 . . . thus giving Cal and his programming staff a good headstart by virtue of their work with the PDP8. The fuel-oil-dealer package was mostly written by Infotecs president Ed Tolson, and you really have to see it to believe how complete it is . . . right down to providing a printout of a customer list showing the overall profit made on each customer.

The program keeps track of each oil truck and its service, sales, route, driver, etc. It bills and sends statements to the customers. It is most complete. Virtually every oil dealer who has seen the system has signed up for one . . . including two in Peterborough.

Infotecs has another program they are just releasing. This one is for accountants; it, too, is most comprehensive. Infotecs buys the printer, the keyboard and monitor, and makes the microcomputer board themselves. The whole system sells to the customer through computer stores or other dealers for about $18,000, which comes to under $350 per month on a lease. At that price, no oil dealer can afford not to get one.

I think Infotecs could sell thousands of their oil-dealer systems if they could make them fast enough and find dealers with enough backing to be able to handle the business. And once they get going on promotion of their accounting system, who knows what could happen?

Of particular note is the disk system used by Infotecs. They're using a PerSci disk with their own operating system. The dual disk holds almost two megabytes . . . that's right, 946,176 bytes per disk. How can they do this? One of the tricks is to split the 12-bit words into two characters of six bits each. They have to forego lowercase to do this, but they can add lowercase if they want to provide a word-processing system later by changing to 12-bit characters.

The CRT has 24 80-character lines, one of the largest video displays in the business. The printer runs 132-character lines, dot matrix, at 125 lines per minute.

I talked with some fuel-oil dealers to see how they felt about the Infotecs system. Those already using it are very enthusiastic. The owner himself is able to do the data input if he wants and thus keep control over his business. The system checks billings against gallons delivered and warns the operator if things don't add up. It also handles such sidelines as diesel oil, gas sales, propane sales, furnace cleaning, etc.

The dealers explained that the Infotecs system is much cheaper

(continued on page 20)

Reader Responsibility

One of your responsibilities, as a reader of *Kilobaud*, is to aid and abet the increasing of circulation and advertising, both of which will bring you the same benefit: a larger and even better magazine. You can help by encouraging your friends to subscribe to *Kilobaud*. Remember that subscriptions are guaranteed—money back if not delighted, so no one can lose. You can also help by tearing out one of the cards just inside the back cover and circling the replies you'd like to see: catalogues, spec sheets, etc. Advertisers put a lot of trust in these reader requests for information. To make it even more worth your while to send in the card, a drawing will be held each month and the winner will get a lifetime subscription to *Kilobaud*!

kilobaud™

Kilobaud is published monthly by 1001001, Inc., Peterborough NH 03458. Subscription rates in the U.S. and Canada are $15 for one year and $36 for three years. In Europe: Kilobaud erscheint montatlich bei Fachzeitschriftenvertrieb Monika Nedela, 7778 Markdorf, Markstr. 3. Abonnement DM 70,- plus Porto 7.20 + gesetzl. MWST; Sfr 81,- plus Porto 7.20. Australia: For subscriptions write—Katherine Thirkell, Sontron Instruments, 17 Arawatta St., Carnegie, Vic. 3163 Australia. UK: £20 sterling/year—LP Enterprises, 313 Kingston Rd., Ilford, Essex, IG1 1PJ, England. Please write for other foreign rates. Second-class postage paid at Peterborough NH 03458 and at additional mailing offices. Publication No. 346690. Phone: 603-924-3873. Entire contents copyright 1978 by 1001001, Inc. INCLUDE OLD ADDRESS AND ZIP CODE WITH ADDRESS CHANGE NOTIFICATION.

EDITOR'S REMARKS

John Craig

Finally: An Affordable Computer-Portrait System

You may have noticed my enthusiasm for computer-portrait systems in previous comments I've made about them. I've always felt that the "hobbyist" community could, and would, come up with something that would sell for considerably less than the $20,000 to $25,000 such systems are currently going for. Well, it finally happened . . . and the Micro Works in Del Mar CA is the company that did it. The photograph in the January issue of the Mona Lisa being generated on a Malibu Design printer caught their eye . . . and that's the printer they selected for the system. Photo 1 will provide a glimpse into what the "Micro Workers" are up to. We'll see if we can get the full details, along with some spectacular portraits, in next month's issue.

Photo 1. The Micro Works system; TV camera on the left, SWTP system; Malibu Design Group printer.

Cybervision Update

Last month I commented on Montgomery Ward's new entry in the home-computer field . . . and said I'd try to get more details for this month. I'm happy to report that I not only got the details; but it looks like we're going to have a review of the Cybervision in an upcoming issue of *Kilobaud*.

I really should point out that it wasn't the hardware that turned me on about this system. After all, it sells for $400 and only consists of a box (with a reset switch), a cassette recorder and two inexpensive touch-pad keyboards. No . . . what excited me was that it was *the first* home system to be introduced to the American public through one of the major mail-order catalogues, and I feel that's significant. Also, it occurred to me that this might be an ideal system for the hobbyist who hasn't made the plunge. The Cybervision's price is low; software is available that the family can put to use right away (for education and entertainment); expanding it in several directions is fun and challenging.

The system is being manufactured for Montgomery Ward by Associated Sales in Columbus OH. After talking to Associated Sales' chief engineer, Jim McConnel, and Joe Miller, their systems programmer, my enthusiasm has certainly increased. Jim filled me in on some very interesting hardware details, not the least of which is that the Cybervision is built around the RCA 1802 microprocessor. My first reaction was, *why?* He pointed out that because the 1802 is a CMOS chip it not only has low power consumption (and other features) but also generates very clean waveforms, which results in negligible generation of radio frequencies. Because of this, getting FCC approval was much easier . . . and the unit doesn't require shielding.

Joe Miller noted a really fascinating item regarding the software. The system only comes with 4K of memory; but because of the use of overlay techniques, programs larger than 4K are run with ease. It takes about eight seconds to load a 2K program segment (via their 2000 baud cassette interface), and such transfers are hidden, or masked, from the user by voice prompts or responses taking place (on the second channel of the recorder) during the transfer.

The Cybervision has other neat features, but let's save them for the review. A refreshing comment from Associated Sales was that they don't consider it a threat to have other people and companies building add-ons and peripherals and generating software for their system. As a matter of fact, they're going to be making details of their bus and interfacing information available for that upcoming article—and their own software development interpreter will be made available in the future. (By the way . . . Tom Pittman will produce a Tiny BASIC for the system in the near future.)

"The Colonel" Goes AWOL

Notice: Norman Henry Hunt (alias Colonel David Wintrop) has pulled off another con. Hunt cut through a fence at Chino (CA) State Prison, where he was doing time for grand theft and fraud (see Editors Remarks, Nos. 10-13), and escaped. He may be setting up shop again. Next month, we'll have a photo of Hunt plus information from the detective who arrested him.

AROUND THE INDUSTRY

John Craig

The Noval 760: Here it Comes!

Do you remember the Noval 760? It first appeared in a full-color ad in the June 1977 issue of *Byte* . . . and captured the imagination of the entire industry. The system is unique, and one of the big reasons is its "packaging." Since a picture is worth a thousand words (I just made that up), I won't strain my typewriter trying to describe the beautiful desk the system comes in—just take a look a Photos 1 and 2.

Actually, that desk is the reason why you haven't been hearing too much of the Noval 760 in recent months. They've gone through *three* suppliers in an attempt to get it manufactured to their specs. The hardware and software was debugged long ago, but the desk has kept them from accepting orders and advertising the system.

The additional efforts have certainly been worthwhile. You know, that computer doesn't just *pop* up out of that desk. Instead, it rises up slowly and gently and brings forth a lump in your throat much as the playing of the National Anthem! (Maybe that only happens with red-blooded computer nuts . . . and not everyone.)

The Gremlin Connection

With the problem Noval has been experiencing with the desk, it's not too farfetched to imagine they might not have survived. However, they are a sister company of Gremlin Industries, and the necessary support has been, and will continue to be, with them (thank goodness for big sisters!). Gremlin Industries is one of the leading manufacturers of electronic arcade games; and the

Photo 1. Is it just another nice-looking desk? No, it's a Noval home system . . . in its "sleeping" position.

Noval 760 is an outgrowth of their efforts in developing microprocessor-based video games.

A few years ago, someone popped up and said, "Hey, why don't we put together a computer system with all this know-how we have?" One of the fantastic bonuses you get with the purchase of a Noval system is the availability of their video arcade games to run on the machine. (That's right, they're the same games you have to pay a quarter to play down in your local pub or arcade!) Photo 3 illustrates the system's graphic capabilities in one of Noval's most popular games, Depth Charge.

The games, and graphics capabilities, offered with the 760 are an important part of the overall system approach. The system is a *home computer*, aimed at the hobbyist as well as the lay user. Since most home-systems buyers have entertainment applications in mind, you can appreciate Noval's emphasis in this area. And . . . you'll be hard pressed to find interactive video games such as theirs on any other systems.

I feel that just as much emphasis should be placed on educational programs as games—combining them is even better—so I was quite pleased to discover Noval has done just that. They're involved with the San Diego School District in a research program called Telemath, and they've developed some outstanding educational games for grades 2 through 6. The games make extensive use of graphics and generate a lot of initiative and competition when played by two students . . . since the responses are timed and the person with the fastest (and correct) answer wins.

A "People" Company

When I arrived at Noval/Gremlin, the first thing to greet my eyes was an *enormous* 40-foot banner spread across the front of the building. "Happy Birthday, Lonnie Pogue," it proclaimed. I thought that was neat. I don't think you can appreciate the significance of the sign, and how it reflects the attitude of the company, until you stop and ask yourself if something like that could happen *where you work*. Probably not, right?

They do some other "strange" things at Noval that you won't find at most companies. For example, they have such a dedicated group that it is not uncommon for individuals to become so enthusiastic and engrossed in what they're doing that they contribute a lot of personal time to completion of projects. This may not be too hard to understand when you consider, after all, that one of their primary products is games. And the fun and games are certainly in evidence. The Gremlin arcade games are set up *everywhere* . . . and available for the employees to play with in their off time, as you can see in Photos 4, 5 and 6.

This "people-oriented" attitude is reflected in the design of the Noval 760, also. The system software was designed with the average hobbyist in mind . . . not the professional program-

Photo 2. The 760 . . . "awakened," and ready for action.

Photo 4. The employee lounge at Gremlin.

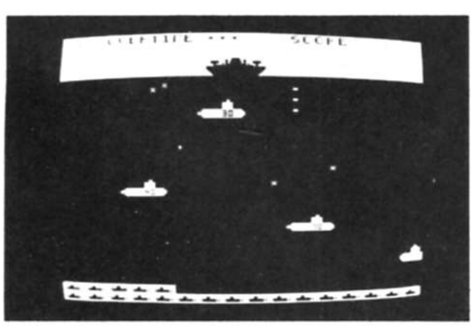


Photo 3. Depth Charge, one of the more popular arcade games available for the Noval 760.

Photo 5. It's really rough . . . games everywhere!

Photo 6. In the development lab new games must be tested . . . in the name of engineering research.

Photo 7. The heart of the 760 system.

mer. It's forgiving, but not sloppy; it's sophisticated, but not complicated; and it's capable of doing serious home software development—within certain limitations (due to a cassette, rather than disk operating system).

The 760 Ingredients

The original design was an 8080-based system, which has since been upgraded to a Z-80. Photo 7 shows the rear of the 760 opened up to expose *the* system board (the whole ball of wax is right there). In the foreground (left front) is the fully socketed 32K of memory that comes with the system. To the left, plugged into the three-connector backplane, is the 760 BASIC in PROM. Additional RAM or PROM memory segments can be plugged into that bus. On the far right, in the back, are the various interface cards; the video graphics circuit is situated in front of them on the main board. So much for the main board—now let's see what goes with it.

The peripherals mounted in the console consist of a 2500 bps digital Phi-Deck cassette unit, a 12 inch b & w monitor (color optional), a 32-column dot-matrix printer and a full ASCII keyboard. I don't know if there are future plans to add another cassette drive, but a 760 with dual minifloppies is on the drawing board. Larger printers can also be ordered, at additional cost. Prices for the 760 start at $3385 . . . and include all the hardware I've mentioned.

The software provided with the system consists of a monitor program, a text editor and an assembler. I found all three easy to use . . . and quite adequate for software development. Just to give you another example of the human-oriented approach in the 760 design, you can use an illegal instruction mnemonic when assembling a program. The assembler will flag you that it is an error; but if you insist on leaving it in, it will simply be changed to a NOP. Another nice feature is the listing of the Editor commands on the screen when you call the Editor up for use.

BASIC is extra. The additional cost (under $300) will get you Noval BASIC or Noval Extended BASIC installed in PROM . . . with an added bonus of freeing up the RAM memory it would normally occupy. Their BASIC allows full interaction with assembly-language programs (loading and executing), output and input from individual I/O ports and (here's the big plus) easy-to-use color and b & w graphics commands. There aren't any string-handling functions in the "standard" version, but they are available in the extended.

(continued on page 20)

Photo 8. Ago Kiss, Noval VP and general mgr.

Photo 9. Noval's research and development lab.

TRS-80 FORUM

Dave Lien

How to Begin Something like This?

By way of introduction, I'm the culprit who wrote the operating manual for the TRS-80. Having been part of the project from its earliest days, I have some small acquaintance with the system. Perhaps in a future Forum we can wax nostalgic and talk about entertaining and humorous events that are part of the project's history.

I work as a college administrator. Hobbies include computers (obviously), writing (maybe not so obviously), ham radio, flying, sailing; and I keep busy consulting. I have no special ax to grind with or for Radio Shack.

The important person in this Forum is you. I'd really rather be off running a computer, but John Craig has a mean hammerlock.

And Away We Go . . .

What do you want to see in the TRS-80 Forum? It's very important that you let me know. Let's try the following format and see what happens.

Flashes from Fort Worth

You aren't alone out there! TRS-80 sales are very strong. It's quite a surprise that about a third of the Level I units are being ordered with 16K of RAM. That was a real sleeper.

Read it and weep! Field failure rate is running a very low two percent—not counting, of all things, power-supply failures. Thought that kind of problem was solved around 1932. Seems some well-meaning soul substituted fast-blow for slow-blow fuses in one batch of the sealed units. Anyway, the problem's supposed to be solved now, and units being delivered have the right fuse. Much worse things could have gone wrong.

All "factory type" inquiries regarding the TRS-80 should be addressed directly to Hugh Mathias, customer service manager. You can call him directly at (817) 390-3583. Hugh and his crew do yeoman service, and I think the enlightened attitude that RS is taking towards customer service is going to go a long way towards making the TRS-80 *the* world's all-time best-selling computer.

Best User Program of the Quarter

There is a classic computer printout from the "heavy frames" that goes back many years: Snoopy shaking his fist at the Red Baron. A microcomputer version of that program simply has to be added to each of our software libraries.

Tom Kasper of San Diego,

one of the first TRS-80 owners, gets software honors this time around for his version of Snoopy in RS shorthand with full TRS-80 graphics. Though we won't comment on his programming style (and we're not going to be too fussy about that sort of thing with any contributors unless some major improvements are mandatory), we think his final product on the screen is just superb! Beautiful job, Tom!

Why not submit your own favorite original program (on any subject)? (Don't worry about not being a professional programmer.) Your fellow TRS-80 owners are hungry for good programs, and will probably "massage" it a bit to suit their own fancy, anyway. Never did see two programmers who completely agreed on how *best* to write a program.

We'll be able to publish *short* routines in the Forum, but, generally speaking, lengthy programs (such as Tom Kasper's) should be submitted as articles. Be sure to send your contribution (to Peterborough) on cassette tape; a hard copy would also be helpful. Record it several times on the tape and include a tape mailer and two first-class stamps if you want the cassette back.

Computer-Scanning the Future

Level II BASIC is not far away. It is nearly finalized at this writing and looks very good. While incorporating the most valuable of the standard Microsoft features, it also retains most of the Level I features.

The cassette Data flow rate has been nearly doubled, and the built-in editor is excellent—similar to the SOS editor used on big DEC machines. Several printers are available but require use of the extra card box for interfacing.

Level III (for use with floppy disks) is close behind, and is a simple upgrade from Level II. Matter of fact, most of what's needed for Level III is already included in the Level II interpreter.

More on the future as we get closer to it.

Feedback from the Field

That's you! Since this is the first Forum, there is only one item of major interest from "the field." Only you can solve that little dilemma.

Of special value will be items of broad interest to other TRS-80 owners: little software or hardware tricks you've discovered, questions of common interest, complaints or just plain comments. The sooner you respond, the sooner we can share your feedback with others. Please write (don't phone): 8662 Dent Drive, San Diego CA 92119.

Dave Waterman of Alpine CA has contributed this valuable feedback. It is a modification to the SINE subroutine found on page 218 of the *TRS-80 User's Manual*. To confirm that a problem really exists, enter the subroutine as printed plus the following lines:

```
10 N = 179.95
20 X = N
30 GOSUB 30380
40 PRINT N,
50 N = N + .001
60 GOTO 20
        and RUN
```

The crash of the subroutine is due to the extremely tiny numbers encountered as the value of the SINE of 180° (and multiples of it) is approached. Dave's solution is shown in Fig. 1 and seems to solve the problem without introducing any new ones. Why not let the other readers of the Forum know how it works for you? Meanwhile, we'll see how Dave's new subroutine works in a really super Level I biorhythm program he is working on, which should appear soon.

```
30000 END
30370 REM  *SIN* INPUT X IN DEGREES. OUTPUT Y.
30380 IF X>360 T. X=X/360 : X=(X-INT(X))*360
30390 IF X>90T.X=X/90:X=INT(X*1000+.5)/1000:Y=INT(X):G.30394
30392 G.30400
30394 X=(X-Y)*90:ONYG.30410,30420,30430
30400 X=X/57.29578 : G.30440
30405 G.30440
30410 X=90-X : G.30400
30420 X=-X : G.30400
30430 X=X-90 : G.30400
30440 Y=X-X*X*X/6+X*X*X*X*X/120-X*X*X*X*X*X*X/5040
30450 Y=Y+X*X*X*X*X*X*X*X*X/362880 : RET.
```

Fig. 1. Dave Waterman's solution.

All things considered, the first printing of the user's manual came out pretty well, but even small typos cause great distress in a computer program. By far the greatest initiator of cards and letters is the typo in the Combined Function and ROM Test on page 227. Line 330 should read:

```
330.F.X=0TOA : A(X)=Q : N.X
```

It seems that thousands of users thought they had bad machines because the factory test program said so. Well—it shows that users really are reading the manual.

More on these and other things in a later Forum.


```
3 REM SNOOPY BY T.N. KASPER, 5 JANUARY 1978.
5 CLS
10 F.A=14TO22:S.(A,0):N.A
20 F.B=12TO15:S.(B,1):N.B
30 F.C=21TO24:S.(C,1):N.C
40 F.C=40TO45:S.(D,1):N.D
50 F.E=10TO12:S.(E,2):N.E
60 F.F=21TO25:S.(F,2):N.F
70 F.G=36TO39:S.(G,2):N.G
80 F.H=45TO47:S.(H,2):N.H
90 F.I=9TO10:S.(I,3):N.I
100 S.(12,3)
110 F.J=23TO35:S.(J,3):N.J
120 F.K=45TO49:S.(K,3):N.K
```

Tom Kasper's Snoopy listing.

(continued on page 20)

INTERTEC's

INTERTUBE™ VIDEO DISPLAY TERMINAL

Something for Nothing!

Intertec's new INTERTUBE™ video terminal is really something! But most amazing is that it costs you almost nothing! The INTERTUBE's $784* price tag combined with our innovative microprocessor design sets new standards in the CRT industry.

The new INTERTUBE obsoletes dumb terminals and out performs the smart ones. Every INTERTUBE boasts such features as a full 24 line by 80 character display, 128 upper and lower case ASCII characters, reverse video, complete cursor addressing and control, a 14 key numeric key pad, 16 programmable function keys, blinking, a self-test mode, protected fields and much, much more.

The new INTERTUBE video terminal from Intertec represents the latest advances in microprocessor technology. Speed, silence and flexibility, coupled with the many operator-oriented features of the INTERTUBE serve to boost the efficiency of both software and programmers.

Our combination of price and performance gives a true advantage to a broad cross-section of users in the computer industry. Whether your requirement is the most sophisticated data entry application or a simple inquiry/response environment, the INTERTUBE is the best value for your terminal dollar.

The INTERTUBE offers a variety of interfaces including RS–232C and 20MA current loop. Better yet, the INTERTUBE interfaces beautifully with the user. Painstaking human engineering assures you of many hours of enjoyable operation without operator fatigue. INTERTUBE's high resolution CRT monitor provides the sharpest possible display image. And our special Accu-Dot™ focusing technique produces ultra clear characters on a non-glare screen.

Our INTERTUBE not only offers unparalleled price and performance but is also backed by Intertec's nationwide factory trained service network providing over 250 local dealer and service center outlets.

There are lots of video terminals on the market today but only the INTERTUBE delivers significantly better performance for such an incredibly low price. So before you buy one of those dumb terminals or one of those other smarter ones, you owe it to yourself to sit down in front of an INTERTUBE. While we admit our motives for recommending that you try the INTERTUBE are purely selfish, it's the only true way to appreciate the design features and price advantage of our new terminal. Mere words just can't do justice to the INTERTUBE. Contact your local dealer today and discover just how smart you'll be when you spend only $784 for your new INTERTUBE.

For the name and address of your nearest dealer, contact us at one of the numbers below. It's the only smart thing to do.

***$784 is INTERTUBE's "Quantity One—Immediate Delivery" price. Dealer inquiries are invited.**

INTERTEC DATA SYSTEMS

Corporate Headquarters
1851 Interstate 85 South
Charlotte, North Carolina 28208
704/377-0300

Eastern Regional Marketing
19530 Club House Road
Gaithersburg, Maryland 20760
301/948-2400

Western Regional Marketing
17952 Sky Park Blvd
Irvine, California 92714
714/957-0300

INTERTEC's

INTERTUBE™

TERMINAL SPECIFICATIONS

MEMORY SIZE:	24 lines by 80 characters per line.
ALPHANUMERIC CHARACTER SET:	Generates all 128 upper and lower case ASCII characters.
LINE DRAWING CHARACTER SET:	Eleven special graphics symbols.
DISPLAY PRESENTATION:	Dark characters on a light background. (Display is reversable through keyboard selection.)
VISUAL ATTRIBUTES:	Blinking (2 frames per second), underline, reverse video, half intensity.
SPECIAL ATTRIBUTES:	Protected, constant and print-only fields.
STATUS INDICATOR LINE:	25th display line reserved for terminal status messages. (Status line displayed as inverse of normal display.)
SCREEN SIZE:	12 inch diagonal.
REFRESH RATE:	60 frames per second (50 frames per second—Export model).
OPERATING MODE:	Conversational: character at a time transmission. Message: line at a time transmission. Page: full or partial screen at a time transmission.
TRANSMISSION MODE:	Half or full duplex, keyboard and switch selectable.
KEYBOARD:	Standard teletypewriter-compatible layout plus 14 key numeric pad, local mode and erase. Keyboard lock/unlock under program control. Also, 16 programmable function keys.
SELF—TEST MODE:	Self-diagnostic firmware testing routine-results displayed on status line.
ADDRESSABLE CURSOR:	Direct positioning by either discrete or absolute addressing.
CURSOR CONTROL:	Up, down, forward, backward and home—keyboard selectable.
READ TERMINAL STATUS:	Allows CPU to interrogate for terminal status, present cursor address and memory value at cursor position.
PARITY:	Choice of even, odd, marking or spacing—keyboard and switch selectable.
WEIGHT:	Approximately 37 pounds.
PHYSICAL DIMENSIONS:	14 5/8" (H) x 21 3/8" (W) x 23 1/8" (D).
ENVIRONMENT:	Operating temperature between 0^{o} to 50^{o} C, Storage 0^{o} to 85^{o} C, 10 to 95% relative humidity, non-condensing.
COMMUNICATIONS INTERFACE:	EIA RS—232C/CCITT V.24 operates through the range of 75 to 19.2K BPS, switch and keyboard selectable. Available 20/60 MA current loop operates through the range of 75 to 9600 BPS.
AUXILIARY INTERFACE:	RS—232C/20MA current loop printer port.

The most remarkable specification of the INTERTUBE is its astonishingly low price. The feature-packed INTERTUBE compares with terminals costing more than three times as much! Combine this price with the inherent reliability of our microprocessor design and the result is an industry first!

$784*

***QUANTITY ONE!**

You don't have to buy a hundred terminals or sign a contract to qualify for INTERTUBE's low $784 price tag . . . it's our "quantity one—immediate delivery" price. Additional discounts are available under OEM, GSA/Educational and Distributor pricing schedules. Contact an Intertec Marketing office for details.

The INTERTUBE and other Intertec products are distributed in these cities: Albany, Alberquerque, Atlanta, Baltimore, Boston, Buffalo,Charlotte, Chicago, Cincinnati, Cleveland, Columbia, Columbus, Dallas, Dayton, Denver, Detroit, Hartford, Huntsville, Houston, Indianapolis, Jacksonville, Kansas City, Los Angeles, Louisville, Madison, Memphis, Miami, Milwaukee, Minneapolis, Nashville, New Haven, New Orleans, New York, Oklahoma City, Omaha, Philadelphia, Phoenix, Pittsburgh, Portland, Providence, Raleigh, Richmond, Rochester, Sacramento, Salt Lake City, San Antonio, San Diego, San Francisco, Seattle, St. Louis, Syracuse, Tampa, Tulsa, Virginia Beach, Washington, DC. **International offices** located in Australia, Austria, Canada, Denmark, England, France, Germany, Holland, Switzerland.

LEGAL/ BUSINESS FORUM

Kenneth S. Widelitz
Attorney-at-Law

Personal Privacy

A discussion of personal privacy and computers looks at only that portion of personal privacy relating to records and record-keeping practices. This comprises a very small portion of what is ordinarily referred to as personal privacy.

The Constitution does not mention a right to privacy. Nevertheless, various aspects of personal privacy have been protected against government action by traditional interpretation of the Bill of Rights. The First Amendment guarantees the rights of free speech, freedom of assembly and freedom of religion; the Third Amendment prohibits quartering soldiers in your home; the Fourth Amendment prohibits unreasonable searches and seizures; and the Fifth Amendment protects against self-incrimination.

Courts have interpreted these protections to prevent an individual from being forced to reveal political, social or philosophical beliefs. Furthermore, the Supreme Court has indicated that individuals have a right to privacy relating to their freedom to practice contraception, have an abortion or read pornography at home. The judicial system has also developed principles that allow suits for invasion of privacy in certain situations involving financial or reputational injury of one person by another.

The foregoing aspects of privacy have nothing to do with the notion of invasion of personal privacy by computer. "Invasion of personal privacy by computer," of course, does not mean the computer is responsible for invasion of an individual's personal privacy. It would be more accurate to say "invasion of personal privacy by *use* of a computer."

It is difficult to define personal privacy in terms of potential invasions by use of a computer. Before we go on, let's speculate on some repugnant uses of a computer to invade what we commonly consider our domain of personal privacy.

You are being interviewed for a job. After the usual meetings with key executives, and the three-martini lunch, a clerk in the personnel department makes a critical phone call to International Data Base, Inc., which reports that although you have not had any prior criminal convictions or arrests, you have ordered \$500 worth of merchandise in the last year from a firm specializing in pornographic films.

The report shows that you bought an additional \$200 worth of materials from an adult bookstore, and made a \$1000 contribution to the Communist Anti-defamation League. It reveals that you travel without your wife on business trips, but always register at hotels as Mr. and Mrs. It also notes that during your sophomore year in college you failed a computer-programming course. During the last five years, the report discloses, you have paid \$10,000 to a psychiatrist. You are surprised when you do not get the job.

To my knowledge, such dossiers are not currently available to potential employers. However, virtually all the information from which such a dossier could be prepared is presently in a computer data base somewhere.

International Data Base, Inc., does not yet exist, but many credit-reporting agencies do. Financial institutions rely on such agencies to provide information. It is commonplace to read of persons who have been denied credit because of inaccurate reports issued by credit bureaus. The cause of such inaccurate reports is uniformly attributed to "computer error."

Back to our definition of personal privacy. From the above, it can be seen that the invasion of personal privacy by use of a computer relates to the compilation and dissemination of information. Some information about you is public, available to anyone to research and use (e.g., records of your birth, marriage, criminal convictions, ownership of real property, address, etc.). Countless other records (educational transcripts, military records, employment personnel files, etc.), though not "public," are accessible by numerous other people.

Even if you shudder at the thought of a firm such as the fictional International Data Base, Inc., consider that every piece of information in their hypothetical dossier on you is already known to many other persons (the pornography company, the Communist Anti-defamation League, the airline-reservation clerk, the hotel desk clerk). You did not think in terms of invasion of personal privacy when you placed the porn order, gave the contribution, made the airline reservation or checked into the hotel.

You can see that in defining the concept of personal privacy with relation to records and record keeping, it is not so much a concern that others know information about you, but rather to what extent that information is compiled and communicated to persons other than those who garnered the data on a first-hand basis—that is, those persons necessary to consummate a transaction. In this context, the right to privacy involves the right of an individual to decide for himself when and on what terms his acts should be revealed to the general public.

As stated by Charles Fried in an article entitled "Privacy" (*Yale Law Journal*, 1968, p. 482), "It is not true, for instance, that the less that is known about us the more privacy we have. Privacy is not simply an absence of information about us in the minds of others; rather it is the *control* we have over information about ourselves."

Of course, for any individual, privacy as a value is not absolute. Its importance can vary with subject matter, time, age, etc. It must also be recognized that the right of personal privacy can conflict with the interests of society (i.e., records maintained for criminal investigations or national-security problems). A conflict also exists in the area of free speech and the public's "right to know."

The Problem Historically

Having ruminated about the definition of personal privacy and the right to it, vis-à-vis computers, it might be useful to consider from where our notions of the right to control information about ourselves come.

From the point of view of the individual, the right to personal privacy in terms of control of information about ourselves is a relatively recent phenomenon. Man has kept records of births, marriages, deaths, etc., almost since he learned to write. They were kept on a very local basis because no methods were available for communicating their content. Virtually no other information about an individual was recorded. It wasn't necessary to ask a credit-reporting agency if a person was credit-worthy. Everyone in his village knew that he was or wasn't.

Perhaps it was with the invention of the Gutenberg printing press, which made the widespread dissemination of information possible, that the question of the existence of a right to personal privacy was first raised.

Subsequent technological advances resulted in the rapid communication of information over long distances. Telegraph, telephone and radio made the world smaller and information on persons outside our communities more important. That technology gave us the means to invade the personal privacy of a limited number of individuals, who tended to be the "important" people, public people, such as politicians, actors and actresses and community leaders. Although it was possible prior to the computer age to accumulate information on individual nonpublic persons, the amount of effort involved in terms of compiling such information was relatively large.

With the advent of the computer, the task of compiling enormous amounts of information on an enormous number of people became possible and economically feasible. Once the capability existed, the applications followed closely behind. It is with the applications, such as credit reporting, that the issue of the right to personal privacy and its invasion comes to light.

Historically, the social utility of record-keeping practices has seldom been questioned. Certainly census information and other statistical records, along with some intelligence (i.e., police, CIA) and administrative records, are valuable and useful tools that have a positive effect on society.

Privacy and Personal Computers

The foregoing discussion has

(continued on page 21)

KB CLUB CALENDAR

Steve Fuller

Richardson/Dallas/Ft. Worth TX

Here are the highlights of a letter I received recently from Neil Ferguson, president of The Computer Hobbyists Group of North Texas:

One of the oldest computer hobbyist clubs in the country, CHG-NT holds two meetings per month—one for the Richardson/North Dallas area and another for members in Dallas/Fort Worth. Meetings are conducted on the first and third Saturdays of the month at 1 PM unless the dates conflict with holidays. Dues of $7 per year include a subscription to the club newsletter, "The Printed Circuit."

The June meeting will be the club's annual swap meet, known as the "Chip and Dip Fest" in honor of the IC. Sounds great!

For more information on the club and its activities, write R. Neil Ferguson, PO Box 1344, Grand Prairie TX 75051, or call him at (817) 265-9054.

Atlanta GA

George Reeves asks us to announce that the Atlanta Chapter of the SOL Users Society (SOLUS) meets twice monthly, on the first Monday and third Thursday. Consider it done, George.

Incidentally, time and space don't permit publication of club rosters, but thanks for the copy. I'll know who to call next time I'm in Atlanta!

Send your requests for details of the current month's club activities to George at 5002 Crowe Dr., Smyrna GA 30080, or call him at (404) 436-0718.

Pensacola FL

The second Thursday of each month is the meeting date for the North Florida Computer Society, according to Eugene Rhodes. Write to him at 227 Edison Dr., Pensacola FL 32505, or call (904) 453-3844.

Ojai CA

W. P. Dart of 231 Valle Rio Ave., Ojai CA 93023, wants to hear from hobbyists in his area who are interested in starting a computer club. His phone number is (805) 646-5824.

Augusta GA

The CSRA Computer Club meets on the third Thursday of each month at 7 PM in the main auditorium of the Augusta-Richmond County Public Library in Augusta.

Rolston Wilder, 2704 Rosewood Court, Augusta GA 30909, will send you a copy of the club newsletter. For more information, write to Rolston or call (404) 733-8750.

Rochester NY

KIM-1 owners in the Rochester area are invited to join Murray Smith in forming a club. He says he suspects there are at least 20 of you out there, so get in touch with him at 1972 E. Main St., Rochester NY 14609.

Investors Club

The Microcomputer Investors Association, a nonprofit professional group, has been formed to facilitate the exchange of information relating to microcomputers and investments including stocks, bonds, stock options and commodities.

In order to benefit from the experiences of others there is a basic requirement that, at least once each year, each member submit an original article for publication in the association's newsletter.

If you'd like an application, send an SASE to Jack Williams, The Microcomputer Investors Association, 2415 Ansdel Court, Reston VA 22091.

Toronto, Canada

Congratulations to the Toronto Region Association of Computer Enthusiasts (TRACE), which celebrated its second birthday in February.

According to public-relations secretary Ross Cooling, "TRACE began in February 1976 as an informal meeting of ten people interested in personal computing, and quickly attracted followers. Since that time the club has grown to approximately 100 members and about as many casual followers. Nearly half of the members have personal systems of some form.

"The main purpose of TRACE is to foster communication and resource sharing among computer hobbyists and professionals. The meeting format includes one or two talks on microcomputer-related topics, and usually a system demonstration. The club also has a monthly newsletter, group purchasing and a library of product literature, books and periodicals. Other activities include flea markets, exhibitions and a software library."

The club meets at the north campus of Humber College at 8 PM on the fourth Friday of the month and at the Ontario Science Centre on the second Sunday of the month. Newcomers are welcome, and information may be obtained from Ross at 488-3314, or Gifford Toole at 828-9202.

Houston TX

Information concerning activities of the Houston Amateur Microcomputer Club (HAuCC) is available from Clifford Carley, corresponding secretary, PO Box 37102, Houston TX 77036; you may call him at 921-7532.

The club's newsletter, "Nybble," contains schedules of upcoming events, product briefs and computer-related job openings in the Houston area.

Columbus OH

The Amateur Computer Society of Columbus (ACSC) meets at 7:30 PM on the first Wednesday of each month at the Center of Science and Industry.

President Fred Hatfield notes that ACSC is working with the Columbus Public Library to set up a "Computer Corner" where patrons will have access to two Commodore PET computers and an extensive technical library.

According to a brochure from the public library, through "the generosity of the membership of the Columbus Amateur Computer Society, the Public Library of Columbus is now able to offer resource materials to the growing number of computer hobbyists in the Franklin County area. Materials . . . are available for loan at the Business and Technology Division of Main Library, 96 S. Grant Ave."

You can contact the Society c/o Fred Hatfield, Computer Data Systems, 1372 Grandview Ave., Columbus OH 43212, or call (614) 488-3347.

Granger IN

A software/hardware library to support the Apple-I computer has been started here. Interested persons may write to Joe Torzewski, 51625 Chestnut Rd., Granger IN 46530.

Midland/Odessa TX

The Permian Basin Computer Group has scheduled meetings of its two chapters as follows:

The Midland meetings will be held on the second Monday of each month at 7:30 PM in the Student Center on the Midland College campus. Meeting dates are: May 8, June 12, July 10, August 14, September 11, October 9, November 13 and December 11.

The Odessa chapter will meet on the second Saturday of each month at 1 PM in Room 203 of the Electronics Technology Building on the Odessa College campus. Dates are: May 13, June 10, July 8, August 12, September 9, October 14, November 11 and December 9.

Details are available from the Permian Basin Computer Group, c/o John Rabenaldt, Ector County School District, Box 3912, Odessa TX 79760. The phone number in Midland is (915) 697-4607 after 6 PM; in Odessa, (915) 332-9151 Ext. 43 from 8 AM to 5 PM.

Cornwall, England

British hobbyists may obtain information about the Personal Computing Club (PCC) from

(continued on page 21)

NEW PRODUCTS

Two New Mainframes from TEI

TEI, Inc., has two versions of their Computer Mainframe System available. The first is Model MCS-112, a foundation unit based on an S-100 bus system with a 12-slot motherboard. The power supply is rated at 17 Amps at 8 V and 2 Amps at ± 16 V.

The second version is Model MCS-122, a foundation unit also featuring the S-100 bus system with a 22-slot motherboard and a higher rated power supply. Both models are housed in a heavy-duty, vented aluminum cabinet.

The power supplies consist of a constant voltage transformer providing brownout protection and showing high noise immunity of better than 100 db between input and output. The front panels of both fully tested and assembled models include an indicating ac switch and reset switch. Prices are $395 for the MCS-112; $495 for the MCS-122.

For more information, contact CMC Marketing Corp., 5601 Bintliff, Suite 515, Houston TX 77036.

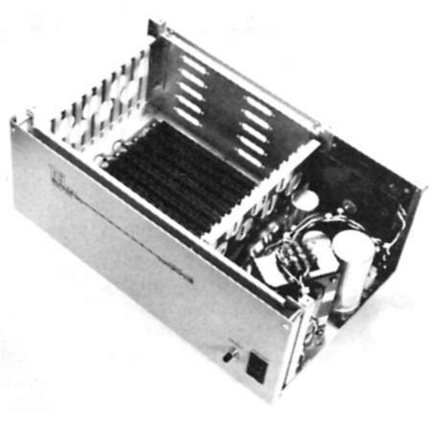
INFO 2000 Systems Software Supported for 8080 and Z-80

INFO 2000 disk-systems owners may now utilize expanded software for both 8080 and Z-80 microcomputers. INFO 2000 Corp. has increased disk-operating software capability using Digital Research CP/M. Among the software packages that operate under CP/M are two full versions of Disk BASIC, including Microsoft Extended Disk BASIC (4.41), $350. Also available is a Structured Systems Group QSORT ($95), a high-speed, general-purpose sort package. A name and address maintenance system, NAD ($79), includes mailing labels. The manufacturer states that all software is currently available for immediate delivery for use on INFO 2000 Disk Systems.

Still available for Z-80 systems is the complete TDL software package—including 12K BASIC, macro assembler, Z-TEL Text Editor and Word Processor—expanded to operate under CP/M and Zapple operating systems. Price, $215.

All INFO 2000 software operates on INFO 2000 Disk Systems that employ the PerSci 277 dual diskette drive with intelligent controller. The Disk System is available for all S-100 microcomputers using Z-80 or 8080 processors, for Digital Group Z-80 and 8080 systems, and for the Heathkit H8.

INFO 2000 Corp., 20630 S. Leapwood Ave., Carson CA 90746.



TEI MCS-112.

New EPROM Programmer

Smoke Signal Broadcasting, PO Box 2017, Hollywood CA 90028, announces a new, low-cost 2708 EPROM programmer. Designated the POP-1, the unit lists for $149 and is designed to interface to the company's P-38-1 and P-38-FF EPROM boards, which are SS-50-bus-compatible products. Complete software is provided on audio cassette. A unique adaptive programming technique allows most 2708s to be programmed in 15 seconds instead of the usual 1½ minutes. A separate, self-contained power supply is used for the programming voltage to insure sufficient current capability to program EPROMs from any manufacturer.

The POP-1.

The Centronics S1.

Microprinter with Serial Interface

Centronics Data Computer Corp. introduces the Microprinter-S1 with serial interface. The new printer, which has a single unit price of $695, allows the user to select baud rates, parity and the number of stop bits.

Centronics expects the S1 version of the Microprinter to enjoy an even broader range of applications than the P1, which debuted last June. Both units are aimed at the home, hobby and microprocessor markets, and both are suited for use in diagnostic systems, CRT hard-copy applications, industrial instrumentation, and demand message printing. However, since many CRT terminals require the RS-232 interface, the S1 should see frequent use as a remotely placed message printer.

Information about numerous other features of the S1 (and P1) can be obtained from Centronics Data Computer Corp., Hudson NH 03051.

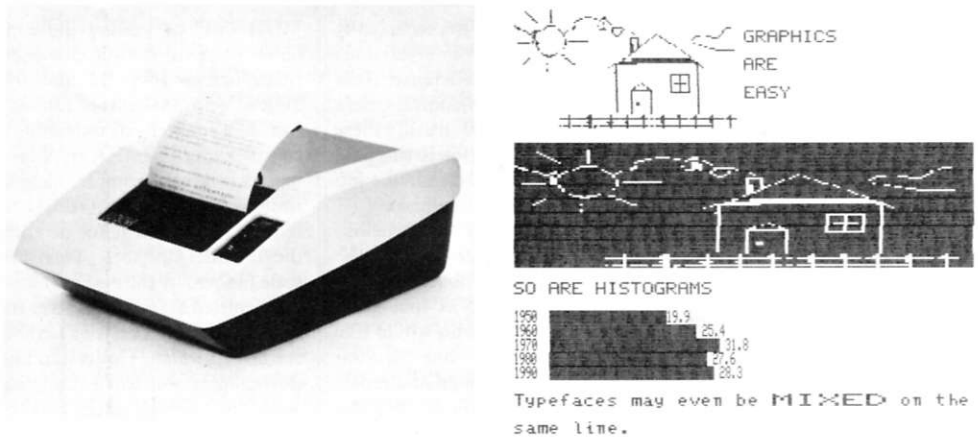
CP/M on North Star Disk

CP/M has become the most widely used S-100 floppy-disk operating system, effectively making it the software-exchange bus for S-100 systems. Vendors of hardware or software supporting CP/M include Imsai, Cromemco, TDL, Digital Systems, Tarbell Electronics, Microsoft, Digital Research and many others. Also, the CP/M Users' Group is currently operating a successful software exchange program.

Now North Star Disk users can also join the software bus. Without any hardware changes, CP/M can be run with all the features available to the users of the system on standard floppy disks. Microsoft FORTRAN-80 and Disk Extended BASIC can also be supplied on a 5¼ inch diskette to run on "CP/M on North Star Disk." All the software is fully 8080/Z-80 compatible.

Retail prices: CP/M on North Star Disk, $112; FORTRAN-80, $400; Disk Extended BASIC, $300. FORTRAN-80 includes relocating assembler and linking loader. Full ANSI except for complex data types. Disk Extended BASIC is a CP/M generation of Altair Disk BASIC 4.1. All prices are freight inclusive.

Lifeboat Associates, 36 W. 84th Street, New York NY 10024.



The Axiom 801 and sample printout from the 820 ($795).

The Axiom Printer

Axiom has a printer, the EX-801, part of the EX-800 series, for fast and inexpensive printing. It prints on aluminized paper ($4 per 230-foot roll) by passing a current through the aluminum coating, evaporating it and thus leaving the black-coated paper underneath exposed. Moisture and sunlight have no effect on the paper. The current is passed through eight wires, making a dot-matrix pattern without the need for moving wires and their fragility.

The printer, with an 8048 microprocessor in control, has an input buffer of 256 characters which can be expanded to 2K, thus providing a full CRT page dump at one time. It will work from RS-232, 20 mA serial as well as parallel ASCII. It comes with 96-character ASCII standard and is expandable to 256 characters with your own programmable fonts. Three character widths, which can be software-selected to provide 20, 40 or 80 columns on the 5-inch-wide paper, can be mixed on a line for emphasis.

Axiom Corp., 5932 San Fernando Rd., Glendale CA 91202.

Audio Cassette Magazine

CLOAD Magazine, published by R D McElroy & Co. of Goleta CA, and written especially for Radio Shack's TRS-80 computer, is something new. It is "printed" on a standard audio cassette and will load directly into the TRS-80 computer. Its "articles" are programs ranging from short games to involved programs of a practical nature; emphasis will be on education.

People can submit programs for publication. There will be 12 issues a year, each issue consisting of an audio cassette with six to ten programs (more, if possible), an index and an instruction sheet. Subscription rates available.

CLOAD Magazine, Box 1267, Goleta CA 93017.

New Business Software Program

A new, low-cost ($200) general-business software package for microcomputers that includes general ledger, accounts receivable, accounts payable, finished-goods inventory control and payroll has been introduced by The Computer Mart, Orange CA.

Developed by Larry G. Grimes & Associates and Computer Products of America, a subsidiary of The Computer Mart, the 24K Grimes Business Information System (GBIS) is specifically designed for small businesses. In a typical application, the GBIS can store on a single minifloppy diskette up to 400 customer listings, 50 vendors, 400 line items of inventory, 25 employee records and 60 general-ledger accounts.

GBIS is written in North Star BASIC, although other disk BASIC languages can be used for the listings. Programs are written so that someone with a minimal understanding of the BASIC programming language can use it.

The Computer Mart, 633 West Katella Ave., Orange CA 92667.

4 MHz Z-80A Processor Board

The ZPB 4 MHz Z-80A processor board is one of several new high-performance products North Star offers for S-100 bus computers. The ZPB will operate in systems with or without front panels. It includes space for 1K of 2708 EPROM.

Other features of this versatile, fast board include auto-jump start-up and vectored interrupts. Available as a kit for $199 or $259 fully assembled. EPROM option costs are $49 for kit and $69 assembled.

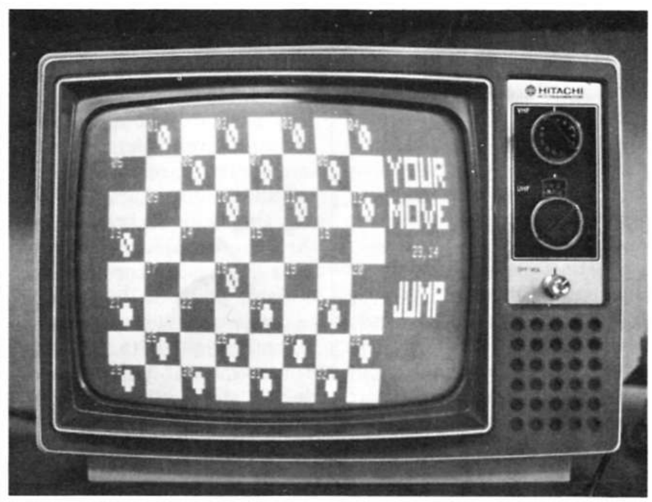
North Star Computers, Inc., 2547 9th Street, Berkeley CA 94710.

Video Checkers on Cassette

Compu-Quote, 6914 Berquist Ave., Canoga Park CA 91307, has developed several games on cassettes, recorded in the Tarbell format and programmed in Mits BASIC (3.1). Contained on one cassette, Video Checkers produces checkerboard graphics on the CRT when used with the PolyMorphic Video Interface and 64-character option. Two versions of the program on one 60-minute cassette play a challenging game that conforms to International Rules. The first version requires a total of 16K of memory, inclusive of 8K BASIC. The second version is more graphic and requires an additional 4K.

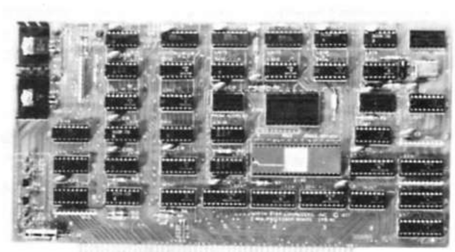
As the player and computer each take turns, the checkers blink and move to indicate their

(continued on page 114)



Video Checkers in action.

CLOAD Magazine.



North Star's ZPB.

BOOKS

An Introduction to Microcomputers Volume 0, The Beginner's Book
Adam Osborne
Adam Osborne and Associates, Inc.
PO Box 2036
Berkeley CA 94702, 202 pages

Hallelujah! At long last, someone has actually produced a book that *really is* for the *absolute novice* in computer technology. Like many books before it, this one claims to be written for people who know nothing about computers. The novelty is that Osborne et al mean it; and what is even better, the book is for the potential user who doesn't know how a computer works, will probably never know and doesn't particularly care. It is written for a pure user orientation, treating the microcomputer as an appliance and recognizing that it is possible to use the appliance without knowing how to design it, modify it or repair it. The reader need only know that the microcomputer exists, that it is potentially capable of performing interesting functions and that he'd like to use one.

Starting from those basic assumptions, the book takes the reader logically and painlessly through a survey of the current state of microcomputers. The first chapter does nothing more than identify the major boxes that make up a microcomputer system and explain some basic data recording concepts. Chapter 2 introduces the book's hero, the indomitable Joe Bitburger. In recounting Joe's experiences with his first basic kit, Osborne describes the frustrations of operating with only front-panel switches and lights for input/output. This brings up consideration of the Teletype and of keyboard entry in general. That consideration leads logically into Chapter 3, which introduces the concept of logical units (files and records), as opposed to physical units, and discusses various components and devices in considerable depth.

Chapter 4 is the obligatory discussion of binary/octal/hex arithmetic and logical operations. There are numerous examples, and the explanations are unusually lucid.

Chapters 5 and 6 deal with software and with system organization. The software discussion defines those mysterious terms—assembler, compiler, interpreter—clearly and concisely. With those concepts established, the chapters describe the various approaches to machine organization. The advantages of different word lengths, of different numbers and arrangements of registers, and other computer elements are clearly explained. The author's intent in these chapters is to bring the reader along to a point where he can intelligently read the specifications of various devices and systems and evaluate these in the light of his own particular requirements.

At least one other aspect of *The Beginner's Book* deserves not only mention, but unrestrained praise: It is readable, even entertaining. The eager, but frequently misguided, Joe Bitburger is an engaging hero. He helps to keep the text oriented to a user (rather than a designer), and to a rank beginner at that. The generous use of graphics and cartoons aids in understanding the text and demonstrates (if it was required) that visual aids and humor can increase the impact of even technical writing.

Less this sound like a commercial for Osborne and Associates, I should mention that the book does have a few flaws. As an example, on pages 5-21, the text refers to a drawing and says that the number representing an instruction code is circled. In fact, there are no circles on the drawing and the referenced number is in a shaded block. This sort of thing can be confusing, but there are few such errors.

A second objection is that the very readable style of the early chapters is not carried all the way through the book. By the end of Chapter 6, Mr. Osborne has basically reverted to the technically thorough but very dry style of his other books. In his defense: He is dealing with more complex data at this point. In truth, there would be no objection to this if he hadn't done such a damn fine job in the first part of the book.

In summary, Mr. Osborne has delivered a first-rate book for the novice computer hobbyist. It is successfully aimed at that great throng of people who would like to use computers but care no more about the design and repair of such devices than they do about the internal workings of their refrigerators. This is a superb book for any beginner and would serve very well as the text for an introductory class at any level from junior high school on up.

Art McDonough
El Segundo CA

Practical Microcomputer Programming: The Intel 8080
Weller, Shatzel, Nice
Northern Technology Books
Box 62, Evanston IL
60204 $21.95

Want to know how your 8080 microcomputer really works? Thinking of writing or modifying system software? Tired of BASIC? Ever wondered about the "Playboy" effect in decimal arithmetic? A necessary step is to learn assembly language, and this book is for you.

Throughout the 18 chapters and 306 pages, the reader is introduced to hardware features of the 8080, presented in a clear, concise, logical manner. Over 80 examples are included, most of which are usable as subroutines to programs you will want to write later. Complete chapters are devoted to such topics as stack pointers, arrays and tables, binary/decimal conversion, communications with terminals, interrupts and subroutines. Sixteen pages are devoted to debugging!

A powerful plus for this book is the authors' determination to demonstrate why and how to use each instruction, not merely to explain how it works. Examples are described first in a short paragraph, then presented in program format with instructions printed in bold letters for easy readability. No matrix-printer-with-a-weak-ribbon printing here!

Reader interest is maintained through occasional use of humor; interchanging memory locations named MOUNTN and MUHAMD or adding the contents of two double-precision fields named INSULT and INJURY help make the learning fun. In a chapter on controlling a complex peripheral (the Victor matrix printer), examples include routines to transform ASCII to required bit patterns for printing the Cyrillic alphabet. This new code (named by the authors) Russian Unified Standard Key for Information Interchange (RUSKII) is then printed. Understanding this chapter will leave the hardware freaks with little fear of tackling an APL printer.

At no point do the authors resort to rehashing material available from the manufacturer of the 8080, but instead choose to take a less theoretical, more practical approach. Oh yes, there is even an index to allow use as a reference manual. One minor objection is that the examples do not strictly follow manufacturers standards for assembly-language programming—labels are not followed by a colon and comment lines begin with a "*" rather than a ";". The style is similar to assembly language used on larger machines.

Any 8080 or Z-80 user would do well to purchase this work, for learning and for reference.

Tim Turner
Spokane WA

Take a Chance with Your Calculator Probability Problems for Programmable Calculators
Lennart Råde, Dilithium Press
PO Box 92,
Forest Grove OR 97116

Take a Chance with Your Calculator will help you understand basic probability theory by running experiments on your programmable calculator.

The book is divided into three sections. The first is composed of eleven chapters containing a total of 143 problems to be investigated, along with some explanations of the problems and the methods to be used in solving them. The second section has 46 pages of more detailed write-ups on many of the problems. Section three has over 100 programs, written for Hewlett-Packard HP-25-type calculators, which are used in solving the problems from section one.

Section one contains the real meat of the book. It starts with a method of generating random numbers and then gives methods of testing random-number generators. These are followed by chapters dealing with the generation of specific types of random numbers, such as the number of spots showing when you throw a pair of dice.

Chapter 5 investigates assorted problems dealing with random numbers, and other problems involved in drawing marbles from an urn containing both black and white marbles. Chapters 6 and 7 deal with towers made of blocks and the results obtained when you move a random number of blocks from one tower to another randomly selected tower.

Chapter 8 examines "runs" that are sequences of the same random digit repeated several times in a row. Chapter 9 is about random walks, when you randomly walk either right or left a random number of paces, and other extensions to this type of walk. Chapter 10 defines statistics terms, including consistency of an estimate, efficiency of an estimate and confidence level, and exhibits test problems.

Finally, Chapter 11 comprises more miscellaneous problems including family planning, the birthday problem, simulation of an election and queuing problems.

Section two consists of comments on the problems in section one. Not all problems have comments, and some comments are quite brief while others are lengthy. The comments may include an explanation of the problem, mathematical formulae, diagrams, flowcharts or hints on how to program a solution for the problem. In many cases, a reference is given to another book where a more detailed treatment of the problem may be found.

Section three has 79 pages of calculator programs and instructions relating to their operation. The programs were written to work on HP-25-type calculators. The book claims they will also run on Texas Instruments SR-56 calculators; but I believe some modification to the programs will be needed since the programs appear to be written for Reverse Polish Notation machines. Texas Instruments uses algebraic entry.

The overall organization of the book is rather poor. I constantly jumped from the problem statement in section one to the comments in section two to the program in section three, then back to section one again. If all of the material for a given problem were collected in one place the book would be much easier to use.

I think the greatest fault with the book is that the author didn't seem quite sure just how advanced his reader would be. In some cases explanations are given in detail, while in others there is no explanation at all. For example, problem 23 asks you to compute the standard deviation. Section two gives a formula for this calculation; but having calculated the standard deviation, nowhere are you told what it represents or what it can be used for!

Even with its faults, there is still a lot of useful material in *Take a Chance with Your Calculator*. The beginner will certainly learn some new concepts; the reader with more background may be given a more intuitive feel for the theory after observing it in practice. On the other hand, if you are trying to learn probability from scratch, you will need a better textbook than this.

Glen Charnock
Oxnard CA

Computers, Computers, Computers: In Fiction and in Verse
D. Van Tassel, editor
Thomas Nelson Inc.
30 E. 42 St., NYC
$6.95, 192 pages

Dennie Van Tassel, author of several successful books for computing professionals, has been gradually moving into broader areas. *The Compleat Computer* (reviewed in *Kilobaud* No. 1) was a tasty assemblage of semitechnical articles, news items, cartoons and some fiction, organized around specific issues of interest to those attempting to understand the effects of computing on our society. Now, in *Computers, Computers, Computers,* he has pulled together a wide assortment of fictional treatments of computers. The result can be enjoyed equally by both computerniks and nontechniks.

The book's 18 short pieces of fiction fall into three categories. The majority (11) are science-fiction short stories by such authors as Robert Sheckley, Gordon Dickson, Robert Heinlein and Barry Malzberg. (In my opinion, Malzberg's contribution, "The Union Forever," is his most successful short piece.)

Another category is satire, and here we find pieces by such luminaries as Art Hoppe, Art Buchwald and Bob and Ray. The third category, poetry, includes 18 computer-tinged limericks ("Glorobots" by Gloria Maxson), as well as several more extended poems and a song ("Push the Magic Button"—to the tune of "Puff the Magic Dragon").

Although the stories, articles and poems can be read and appreciated separately, it seems more interesting to view the collection as a whole. In so doing, try to extract from the different treatments some common notion of what nontechnical people have thought computers will do to us. The title of the author's introduction says it—"Make Way for the Machines."

Rich Didday
Santa Cruz CA

LETTERS

From Don Tarbell . . .

It seems that many new computer customers have been given the impression that it's easy to get a computer system up and going. The truth is that it can be a can of worms, depending on the user, the components and the system required.

There is one thing in particular that many people don't seem to understand: There is a world of difference between a bunch of hardware and software computer components and a computer system up and working and doing what you want it to do. This difference is sometimes big, sometimes small, but it's almost always at least ten times as big as you think. This difference is called system design and integration, and there are companies that make millions of dollars doing things like it. This phase is when you discover that all "S-100 compatible" boards are really not compatible with each other. It's when you find out that two big expensive pieces of software can't talk to each other. It makes people work weeks or months without getting anywhere. In short, you stand to lose your shirt if you don't know what you are doing!

Everyone runs into system integration sooner or later, but it seems to get really nasty in small-business systems. I suppose one reason is that it all sounds so simple! The boards are proven, the software is extensive. Everything required sits right there in front of you. All you need to do is make a few patches here and there and write a little applications software.

Of course, it's those little details that get you every time. Since I get many calls from people about their problems, I'd like to present short answers to a few that seem to recur.

1. Many Z-80 CPU boards don't create a true Altair-bus environment. This means that although the CPU board will work in some situations, with some memory and with some interfaces, it may not work with the combination *you* want.

2. Most 8080 software is written to run at a particular location in memory. If two programs that must run together are written to run in overlapping memory, one must be moved. It is not always easy to move programs, particularly if there is no source available.

3. Most video boards require a good-sized driver program. This driver must be present in memory concurrently with your BASIC interpreter, or whatever, and furthermore, each program that must access this drive (probably all) will have to have references to this driver in it.

4. All dynamic memory boards require a refresh cycle. Even though some manufacturers will claim theirs is done at a time that is "invisible" to the rest of the system, there is no guarantee that some other manufacturer's board won't need to access memory at this same "invisible" time.

5. Some programs that work fine in your main computer memory (RAM) may not be easily modified to work with a mass-storage device such as a floppy disk or cassette. Often a whole new approach is required.

6. Noise is almost always a problem. Some S-100 buses have more noise than others. Terminating resistors don't always help. The third prong on one wall

plug may not be connected to the third prong on another wall plug (would you believe 50 volts of difference?).

7. Be sure that there are no two interfaces that use the same port numbers.

8. There is no real standard on console port numbers, so either software or hardware may need to be modified. Sometimes the hardware is easier.

9. ASCII is not always ASCII, RS-232 is not always RS-232, etc.

10. Some mainframes that don't have a front panel also don't provide some of the requirements of the bus that a front panel would have, such as pull-up resistors, MWRITE line, etc.

11. Because every single component in your system tests good in someone else's system, that doesn't mean they will all work together. The best test is the whole system the component will work in.

12. There is no memory test that goes through all the combinations, regardless of what anyone might say. Figure it out yourself if you must, but such a test would take longer than your lifetime to run. After all, what is 2Λ(4096*8), anyway . . . and then, of course, there's the bit that drops out after time . . .

13. Programs always take at least five times longer to write than you most conservatively estimate, even when the hardware works perfectly all the time, which, of course, it won't.

14. Requirements for software should be specified before those for hardware. Design the system from the top down if possible.

The preceding is not meant to give everyone a crash course in system design, but only to point out that there may be problems you haven't thought of. If you are a hobbyist, you are probably willing to learn (sometimes the hard way). If you're a systems designer, be sure to keep on learning. And if you're a businessman, find someone who really knows what he's doing to help you out. It may cost you as much as you paid for the hardware and software (or more); but, believe me, if you don't have the time or inclination to learn it yourself, the money spent is well worth it!

In the past several months, I've talked to many people who are in trouble now because they underestimated the task they undertook. These people would have been better off if they had retained someone who already made the mistakes a few years ago. The computer stores around the country, especially the older ones, are the best places to find people like this. They've had the experience to know which boards work with which, and which software is useful. If you want to shop around and buy one piece here and one piece there, directly from the manufacturers, you're taking on the job of designing the system yourself. So don't blame the manufacturers or the stores where you bought these separate items for your problems. If you're not willing to take the system's responsibility (and that includes maintenance) on yourself, find someone who will.

Some manufacturers claim to be selling computer "systems." Watch out! Is the manufacturer local? Does he provide a good warranty on the complete system? Will the software he provides really do the job? My feeling is that only the largest manufacturers, such as IBM, DEC and UNIVAC, can afford to provide local support for complete business systems of any size (even the smallest). That's why I've chosen to avoid the word "system" in any of my advertising and refer as much business as possible to a customer's local dealer. I would rather sell mostly to true computer hobbyists, dealers and system designers.

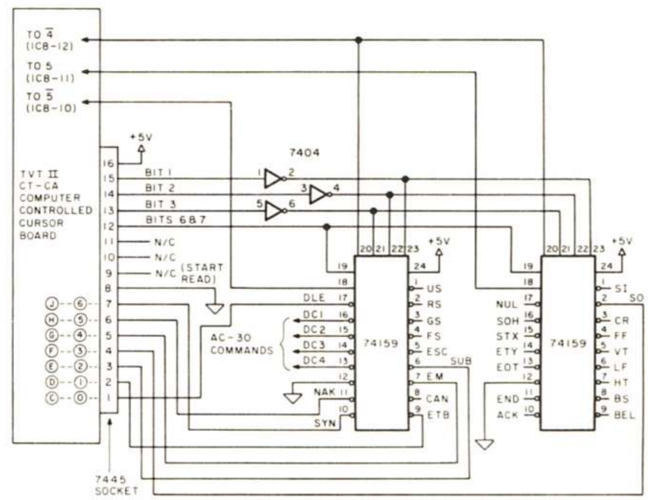
I hope that the above has helped someone. I don't want to discourage anyone from getting a computer system. I use one in my own business and feel that it's one of the most valuable assets I have. I just want people to have enough information to buy at the right place.

Don Tarbell
Tarbell Electronics

Many thanks, Don. This is a problem we've been trying to bring out in the open for some time. Now, if we can just get some articles into Kilobaud *dealing with the problems encountered in system integration . . . the benefits will be tremendous.—John.*

Response to October Article

As a TVT-II owner, I read with interest Richard Wright's article (*Kilobaud*, October, 1977) describing ways of decoding the much needed additional ASCII control codes. The circuit shown in Fig. 1 should not have been dismissed so quickly, though, as the substitution of a 74159 open-collector decoder for the 74154 overcomes the "one flaw" mentioned



by the author. By leaving the TVT-II Cursor Control board in place, removing the 7445 and using the bit signals available at its socket plus the $\overline{4}$, 5 and $\overline{5}$ signals available elsewhere on the board, a 32-character, strobed decoder can be made using only two 74159s and half of a 7404. The enclosed schematic shows the arrangement I am presently using in my TVT-II. The page select, cursor control and outside-world commands are implemented in basically the same manner as shown in Fig. 3 of Mr. Wright's article using this simplified decoder.

Roger Wileman
Escondido CA

Algorithms and Flowcharts, Please

As a KIM-1 owner and graphic programmer by trade, I feel justified in donning my cloak of righteous indignation when confronted with Larry Woods' biorhythm program (*Kilobaud* No. 14, February 1978).

In any program, one must compromise between the two mutually exclusive ideals—small size and quick execution. Although a biorhythm algorithm is trivial, Larry's program is not.

The DAY # algorithm in the algorithms I have provided will work for any date between 1 March 1900 and 28 February 2100. I found it in an old computer journal. As a side effect, MOD (DAY # (D, M, Y), 7) yields the day of the week—SUN :: = 0, MON :: = 1, etc., so why all the tables?

I share these algorithms for two reasons: (1) to try to stem the flood of biorhythm programs being published and (2) to demonstrate the proper way to share algorithms (i.e., in an implementation-independent form). Take note that an algorithm is not a program (all apples are fruit, not all fruits are apples).

Though the exchange of programs should be encouraged, the 6502 owner derives little from a Z-80 chess program, much from the algorithm for collecting the set of squares containing pieces that can be captured by either color.

An algorithm can be programmed in any computer language—BASIC, PASCAL, Z-80 or 6502 assembler. An algorithm is more machine independent than a BASIC program.

I feel cheated by every program that is published in assembly-language for a machine I'm not familiar with—I must try to conjure the algorithm from the code in order to make use of it, and try not to get sick at a KIM-1 program that can't be placed in ROM because it's self-modifying.

But to get back to my original purpose for writing—I feel that the editors must take more responsibility in ferreting kludges from this publication. You could start by requiring that a set of flowcharts accompany every assembler program.

Starting with a published algorithm, rather than a machine-dependent program, each reader can write a program for his own machine that takes advantage of its unique environment (I/O routines, math software, addressing modes).

I don't want to see a CompuCraft micro APL program for the game of checkers with a CyberClops CRT/TTY MK IV, but I might try my hand at programming go-maku for my KIM-1 if I had an algorithm in a structured psuedo-language.

D. A. Harrod
Rochester NY

As a matter of policy we always try to insure that flowcharts are included with software articles. Sometimes we slip up, too. As far as Larry Woods' article is concerned, it works. As a matter of fact, I ran it on the day I received your letter and it said I was at an emotional peak. Your letter took care of that!—John.

```
BIORHYTHM: PROC;
  AGE   = DAY # (TDAY, TMON, TYR)-
          DAY # (BDAY, BMON, BYR);
  PHYS = MOD (AGE, 23)
  EMOT = MOD (AGE, 28)
  INTL =  MOD (AGE, 33)
END BIORHYTHM;

DAY # (D, M, Y)
  IF M > Z
          THEN DO
            K = Y
            L = M + 1
            OD
          ELSE DO
            K = Y - 1
            L = M + 13
            OD
  FI
  K = INT (365.25 * K)
  L = INT (30.6 * L)
  RETURN K + L + D - 621049
END DAY #
```

Innovations from the Hobbyist Community

I have been reading *Kilobaud* since it was first published, and I especially enjoyed the quality of the articles in the January issue. No one publication can serve the diverse needs and interests of the rapidly growing microcomputing /small-systems field. And *Kilobaud* is providing me and other enthusiasts with its unique approach. Let me be specific.

Dr. Michael Wingfield's article, "Hardware Program Relocation," was a breakthrough for the small-system user. There is no reason why many of the sophisticated features that accompany the medium and large-size commercial systems cannot be developed for small systems . . . and Dr. Wingfield's article is more evidence of this. Program relocatability is essential if small systems are to maximize their potential power. Dr. Wingfield's article represents a milestone in hardware development for the microcomputer user, one that was long overdue. One wonders why some of the systems manufacturers don't pick up the ball and contribute to the enhancement of their systems with such "innovations."

Mr. Parsons' article, "Understanding Loaders," was an exceptionally well-written and timely article. His portrayal of a system's software as a series of loaders, one bringing in the next in a hierarchical manner, is a very important concept for users to grasp. One can even say (stretching his analogy to its limits) that the software of a system is only made of loaders including applications programs that load data, subroutines and user commands. Mr. Parsons' fluid writing style enables him to present complex topics in down-to-earth language . . . which all of us appreciate.

These were but two of the many excellent articles in issue No. 13. They represent the kind of quality that I hope will continue to be part of your magazine and from which others with a desire to write can learn. Please pass on my thanks to both of the above authors and keep up the good work.

Dr. Paul R. Poduska
Lowell MA

Objective Review

The January 1978 *Kilobaud* was the *best yet.* The articles "Has Godbout Done It Again?" and "The TRS-80: how does it stack up" are very beneficial to those of us who are deciding on a system but who haven't the knowledge or resources available to make an intelligent decision.

I also found the article "Understanding Loaders" very informative and helpful in my coming to a decision on a system.

Please give the readers more articles concerning product evaluation and performance vs price. Very few of us have other resources (computer stores) for which to make these evaluations. Your *critical* advice will be greatly appreciated by your readers.

Dwight T. Still
Lawrenceville GA

Micro Maestro Modification

I tried Terry Perdue's "Micro Maestro" program in the January 1978 issue. Given the simplicity of the program and the "interface," I was pleasantly surprised by the results. Terry must be complimented on his arrangement of "Somewhere My Love." After playing with it for a while, I became frustrated by the program's inability to play rests. I modified the program slightly to allows rests—a whole rest (the length of a note) by coding $02 into the table, and a "half" rest (one-half the length of a note) by coding in a $01. The change is simple and adds only 21_{10} bytes to the code. It makes my version of "Mary Had a Little Lamb" much more understandable.

I also set it up so the program gets its speed parameter from a table (the *speed* value), instead of writing it into the code. This makes it possible to burn the program into a PROM and still be able to change the speed.

I have tried both these changes, and both are incorporated in the listing shown. As I write this letter, another change, which I haven't tried yet, occurs to me. As the program is written, it plays eight different versions of the song, then repeats the cycle. Actually, it only *plays* seven versions since the eighth has all output bits disabled and, therefore, plays nothing. This silence is annoying in a long song.

I suggest the following modification to cure this. After the INC $8004 instruction, insert these two lines:

```
TST $8004
BEQ REPLAY
```

I think this will eliminate the "silent song."

Well, that's about all. If you haven't tried the musical micro, I urge you to do so. It's neat!

Don Korte
Flint MI

```
                         NAM    MUSIC1
               *
               * M6800 MUSIC PLAYING PROGRAM
               *
               * WRITTEN BY TERRY PERDUE
               *      KILOBAUD JANUARY 1978 P. 94
               *
               * MODIFIED BY DON KORTE
               *      ALLOWS HALF REST, NOTE CODE=01
               *      ALLOWS WHOL REST, NOTE CODE=02
               *
0050                     ORG    $50
0050 0900        SPEED   FDB    $0900    INPUT SPEED PARAMETER HERE
0052 0002        HSPD    RMB    2
0054 0001        TEMP    RMB    1
0055 0001        MUSIC   RMB    1        MUSIC TABLE BEGINS HERE
0000                     ORG    0
0000 7F 0053             CLR    HSPD+1
0003 96 50               LDA A  SPEED
0005 47                  ASR A
0006 97 52               STA A  HSPD
0008 7F 8005     REPLAY  CLR    $8005
000B 7C 8004             INC    $8004    CHANGE HARMONIC QUALITY BY
               *                           BY ENABLING DIFFERENT
               *                           OUTPUT BITS
000E 73 8005             COM    $8005
0011 8E 0054             LDS    #TEMP    POINT TO FIRST NOTE - 1
0014 DE 50       NUNOTE  LDX    SPEED    TIME PER NOTE
0016 33                  PUL B           PULL NEXT NOTE FROM TABLE
0017 5D                  TST B           LAST NOTE?
0018 27 EE               BEQ    REPLAY
001A C1 01               CMP B  #$01     HALF REST?
001C 27 14               BEQ    HREST
001E C1 02               CMP B  #$02     WHOLE REST?
0020 27 12               BEQ    WREST
0022 D7 54               STA B  TEMP
0024 4C          TONLUP  INC A           REGA HOLDS THE VALUE WHICH
               *                           IS SENT TO THE PIA
               *                           COUNTING THESE AROUND GIVES
               *                           HARMONIC QUALITY TO THE OUTPUT
0025 D6 54               LDA B  TEMP
0027 09          COUNT   DEX             COUNT DOWN NOTE TIME
0028 27 EA               BEQ    NUNOTE   DONE
002A 5A                  DEC B
002B 26 FA               BNE    COUNT    AUDIO CYCLE NOT DONE YET
002D B7 8004             STA A  $8004    THIS CYCLE DONE.
               *                           STORE A NEW OUTPUT CODE
0030 20 F2               BRA    TONLUP
0032 DE 52       HREST   LDX    HSPD
0034 7F 8004     WREST   CLR    $8004    CANCEL OUTPUT FOR A WHILE
0037 01          WAIT    NOP
0038 01                  NOP             THEN WAIT
0039 01                  NOP
003A 09                  DEX
003B 27 D7               BEQ    NUNOTE   FINISHED
003D 20 F8               BRA    WAIT
                         END
```

PUBLISHER'S REMARKS

(from page 5)

to use than an outside service system . . . and they have their information when they want it. If someone calls up and wants to know when to expect the next oil delivery, the operator can quickly find this information and tell him. This would be impossible with a service agency. Cheaper and better, a good combination.

Although I was a bit disappointed to find that Infotecs has developed its own programming language, even to the extent of writing its own compiler, this does make some sense from the viewpoint of system integrity. Even if some other firm gets a dump of the programs, they won't make sense without the compiler. It would take an enormous amount of work to break the programs down and use them on anything else. It wouldn't be worth the trouble.

Most of the programming credits go to company president Edward Tolson. It's the programming that makes the big difference . . . and that's a fact.

Interested dealers could do worse than get in touch with Infotecs about distributing this system. Infotecs, 1 Perimeter Road, Manchester NH 03103.

Any Other Systems Around?

As far as I know this is the first microcomputer system that is being produced, complete with programming for a specific industry. This looks to me like an excellent approach, and I'm interested to hear if any other firm has a similar system going.

Algorithmics

A small outfit from Massachusetts has been working hard to come up with a small-business microcomputer system, complete with programming. The other day I got a call from Algorithmics with a request that I come down to the Microcomputer Store in Nashua (NH) and take a look at their system.

They've come up with a very sophisticated word-processing system, using the new Seals PUP microcomputer, a Diablo printer and a keyboard of their own design and manufacture. This has allowed them to design a set of special keys for control functions separate from the regular typewriter and number-pad keyboards.

The system has one of the more comprehensive editing functions I've seen. The printer is a fast job that prints forward or backward and has a nice clean look . . . excellent for business letters. Dot-matrix printers are all OK for bookkeeping functions, inventory dumps, etc., but when you are preparing business letters you have to have a clean impact printer.

The system also does graphics . . . it ground out a 73-point star in a couple of minutes. I've never seen a 73-point star before . . . and how else would you get one than from a computer?

This type of entrepreneurial work is badly needed in our fledgling industry. Dealers everywhere are crying for systems, complete with software. So far there hasn't been very much, and it is a wonder that dealers have been able to keep their heads above water. The demand is there, so let's hope we see more total systems such as the Algorithmics coming along.

Getting Newcomers Started

In adapting to computers, the first steps are the most difficult. Newcomers face a bewildering array of electronic circuits, the whole maze of programming and its languages, all made even more difficult with the barrier of a foreign language: computerese.

Those of us familiar with vectored interrupts and other such nontrivial concepts enjoy jargonizing in our arcane dialect of English.

However, let's be nice to beginners and try not to put them down. But what can we really recommend . . . by now most of us have forgotten just how we got to understand microcomputers and we certainly are unable to keep up with the flood of books. So what can we suggest newcomers do to get relief?

May I suggest you point them to their nearest computer store and a book called *Hobby Computers Are Here*. All the fundamentals are there, written for the beginner. It starts with a chapter on what a computer is, then goes on to how it counts, binary numbers and arithmetic. It explains the basics of computer languages, and then goes into the circuits that make up a computer: gates, flip-flops, etc. This is the ideal first book. It costs $4.95 and, as should be no real surprise, we publish it.

TRS-80 FORUM

(from page 9)

```
130  S.(7,4):S.(8,4):S.(13,4):S.(23,4)
140  F.L=50TO56:S.(L,4):N.L
150  S.(5,5):S.(6,5):S.(14,5):S.(15,5):S.(21,5):S.(22,5)
160  F.M=53TO57:S.(M,5):N.M
170  S.(4,6):S.(16,6):S.(19,6):S.(20,6):S.(21,6):S.(27,6)
175  S.(28,6)
180  F.N=53TO57:S.(N,6):N.N
190  S.(3,7):F.P=16TO20:S.(P,7):N.P
200  F.Q=52TO56:S.(Q,7):N.Q
210  S.(2,8):S.(19,8):S.(50,8):S.(51,8)
220  S.(2,9):S.(19,9):F.R=28TO49:S.(R,9):N.R
225  S.(18,13):S.(25,13)
230  S.(1,10):S.(18,10):S.(27,10):S.(1,11):S.(8,11):S.(9,11):S.(10,11)
240  S.(18,11):S.(26,11):S.(1,12):S.(7,12):S.(11,12):S.(18,12)
250  S.(25,12):S.(1,13):S.(6,13):S.(7,13):S.(12,13)
255  S.(18,13):S.(25,13)
260  S.(1,14):S.(7,14):S.(12,14):S.(13,14)
270  S.(18,14):S.(25,14)
280  S.(2,15):S.(8,15):S.(12,15):S.(19,15)
290  S.(26,15)
310  S.(2,16):S.(9,16):S.(10,16):S.(11,16):S.(19,16)
330  S.(27,16):S.(3,17):S.(19,17)
340  F.A=28TO37:S.(A,17:N.A:S.(46,17):S.(47,17):S.(48,17)
```

AROUND THE INDUSTRY

(from page 8)

The Noval Team

Let me introduce you to a few

of the key people who keep things moving at Noval. You can usually find Agoston (Ago) Kiss, vice-president and general manager, in a working position much like that in Photo 8. Ago is involved in both hardware and software developments, and really put the system through its paces while I was there. (I was told that he came to this country from Hungary in 1956. Because of certain unwelcome "visitors," he had to beat a hasty retreat . . . and crossed over the border carrying a suitcase in one hand and his son George in the other. George is now software manager at Noval and is shown standing in Photo 2.)

Photo 9 was taken inside the top-secret, level-3-badges-only, research and development lab at Noval. Lane Hauck (the Director of R and D), seated on the right, and Bill Blewett are deeply involved in developing, and playing, a new video game. (What a life! I wonder if they have any openings?)

I had a couple of other photos to share with you, but my camera started acting up. When I attempted to use the strobe, the charge for firing it off came out of the camera into my left hand . . . passed through my body . . . and then returned to the camera via my right hand. Felt so good I just had to go back and try it again!

Terry Sorenson, Noval's Chief Engineer, was one of those I missed during my camera misadventures, but he was sure helpful during my visit, the invitation for which came from Jerry Hansen, the president of Noval. All I can say about him is that he's definitely one of the "white-hat good guys." As I said earlier, it's a "people organization." They not only have a knack for making you feel at home during a visit . . . I'll bet they make you feel just as much at home when you do business with them.

LEGAL/BUSINESS FORUM

(from page 12)

related the right of personal privacy to large computers owned and used by various agencies of government and large private organizations. Before, the issue had never been raised in terms of personal computers owned by individuals because such contraptions did not exist. They do exist now, and, I suppose, it will only be a matter of time before personal computerists will be able to purchase all kinds of information about individuals in the same manner they can buy mailing lists today.

The possibility of blackmail by the use of computer-compiled information is not absurd. The currently existing right-to-privacy legislation, which will be discussed in the next Forum, can be applied to personal computers by changing a few words in a definition. Someday, perhaps sooner than we think, we will face the problem of licensing the use of computer technology.

KB CLUB CALENDAR

(from page 13)

H. G. Humphrey, chairman, c/o The Micro-B Computer Store, 22 Lemon St., Truro, Cornwall TR1 2LS England.

If your Club Calendar announcement contains timely, in addition to general, information about your club, please mail announcement at least two months before the date or dates mentioned in the announcement. Please mail all Club Calendar-related material to: KB Club Calendar, c/o Steve Fuller, PO Box 218, Spofford NH 03462.

Attention, Authors!

If you write an article for *Kilobaud*, and if that article contains program listings and/or runs, please submit *camera-ready* listings/runs. Use a printer (borrow one if you don't own one), and please be certain that the machine used produces dark, clear copy (also, try to avoid misspellings). If you must use a typewriter, and if it doesn't have a carbon ribbon, please make sure that it has a relatively new ribbon. Single-space (or, preferably, use space and a half if your machine has this capability) particularly if the listing or run is a long one. This will make our job immeasurably easier. Thank you.

```
350   F.A=51TO54:S.(A,17):N.A
360   S.(4,18):S.(20,18):S.(21,18):S.(22,18)
370   S.(36,18):S.(44,18):S.(45,18):S.(50,18)
380   S.(51,18):S.(54,18)
390   S.(5,19):S.(22,19):S.(35,19):S.(44,19):S.(53,19)
400   S.(6,20):S.(7,20):S.(22,20):S.(33,20):S.(43,20)
410   S.(49,20):S.(53,20):S.(8,21):S.(9,21)
420   F.A=24TO33:S.(A,21):N.A:S.(42,21)
430   F.A=49TO52:S.(A,21):N.A
440   F.A=10TO12:S.(A,22):N.A
450   S.(26,22):S.(27,22):S.(31,22):S.(32,22)
460   S.(41,22:S.(48,22)
470   F.A=13TO25:S.(A,23):N.A:S.(32,23)
480   S.(40,23):S.(47,23)
490   S.(25,24):S.(33,24):S.(39,24):S.(46,24)
500   S.(25,25):F.A=34TO38:S.(A,25):N.A:S.(45,25)
510   S.(26,26):S.(44,26)
520   S.(27,30):S.(33,30):S.(46,30):S.(47,30)
530   S.(38,26):S.(39,26):S.(43,26)
540   S.(27,27):S.(28,27):S.(28,27):S.(39,27):S.(43,27)
550   S.(28,28):F.A=40TO43:S.(A,28):N.A
560   S.(28,29):S.(32,29):S.(43,29):S.(44,29):S.(45,29)
580   S.(27,31):S.(29,31):S.(34,31):S.(48,31):S.(49,31)
590   S.(26,32):S.(30,32):S.(35,32):S.(50,32)
600   S.(26,33):S.(31,33):S.(35,33):S.(36,33):S.(51,33)
610   S.(26,34):S.(31,34):S.(36,34):S.(37,34):S.(51,34)
620   S.(25,35):S.(31,35):S.(38,35):S.(50,35)
630   S.(25,36):S.(31,36):S.(39,36):S.(49,36)
640   S.(25,37):S.(29,37):S.(37,37):S.(38,37):S.(48,37)
650   S.(26,38):F.A+30TO33:S.(A,38):N.A
660   S.(38,38):S.(47,38)
670   F.A=27TO29:S.(A,39):N.A
680   F.A=33TO37:S.(A,39):N.A
690   F.A=44TO46:S.(A,39):N.A
700   S.(30,40):S.(31,40):S.(33,40):S.(41,40):S.(46,40)
710   S.(32,41):S.(33,41):S.(41,41):S.(46,41)
720   S.(33,42):S.(41,42)
730   F.A=46TO56:S.(A,42):N.A
740   S.(33,43):S.(41,43):S.(56,43):S,(57,43)
750   S.(32,44)
760   F.A=41TO51:S.(A,44):N.A
770   S.(56,44):S.(57,44)
780   S.(31,45):S.(32,45)
790   F.A=51TO56;S.(A,45):N.A
800   S.(31,46):S.(32,46):S.(51,46):S.(52,46)
810   F.A=32TO51:S.(A,47):N.A
900   P.A.271," * * * * * * CURSE YOU RED BARON !!!!"
910   P.A.936," TOM KASPER 5 JAN 78"
10000  G.10000
```

Ralph Wells
Vice President Engineering
Inmarco, Inc.
7655 Sunset Blvd.
Hollywood CA 90046

PET's First Report Card

an objective evaluation

Photo 1. A family portrait: PET, center, with granddaddy, Jolt, left, and proud father, KIM, right.

A year and a half ago I wrote an article for *SCCS Interface Magazine* comparing the eight personal computers I had bought, built, designed, redesigned and debugged (or failed to debug). At that time, the PET was only a gleam in its father's (Chuck Pettle) eye. Now I have one.

At the end of October my PET arrived, three and a half months after the usual "$800 cash-up-front" type order that most of the others required. Although two weeks tardy, it had a better record than any of the others, except the Scamp and Tarbell, which arrived on time. I've come to accept late delivery as a way of life for newly announced equipment, but I find that most people entering the exciting and mobile field of personal computing balk at it—especially compared to today, when you can walk into a computer store and get products from maybe a dozen suppliers on an off-the-shelf basis. Furthermore, you pay for them next month on a credit card.

I continually hear the query, "Is it worth it?" It is. The day after I received my PET I took it to a meeting of the Valley Computer Club and was barraged with similar questions from people who already had their own computers. How does it stack up?

This "report card" is an attempt to answer some of these questions on an unbiased basis (I'm not selling anything). I have personally built or bought and modified three 8080-based, five 6800-based, three 6502-based and one SC/MP-based microcomputers, so the PET has a lot to live up to.

Background

In December 1977, Commodore had never advertised the PET, but the magazine articles, television exposure and convention displays made it a pre-production marketing phenomenon. In fact, I assume the reader has already been exposed to its fundamental specifications. In case you haven't, just pick up nearly any back copy of any computer magazine starting last July (e.g., Sheila Clarke's article in the September 1977 issue of *Kilobaud*).

Right up to the present time the big question has been: Can Commodore produce what they claim for the quoted price and still make enough money to stay in business? To get some official answers from them, I wrote a two-page letter and received a one-sentence reply that contained an honest admission to "crummy documentation." After this article was half written, I had a chance to chat with Chuck Pettle, but the opinions herein expressed are my own, derived primarily from personal PETting and augmented by the published references and conversations with sales representatives.

To start with, I'll dive into the deep end of the pool of controversy and say that, in my opinion, they're *going* to make it—and make it big! Not with the model I received (serial 171), but because of *vertical integration* and forward-thinking management.

Setting the Stage

Let's review some history to get a perspective of the pros and cons of grading the PET #171. In a sense, this is more of a mid-term interim report card because the PET's true potential has not yet been adequately documented. I've spent most of my time trying to find out (the hard way) just what I bought. There is a gnawing feeling in the pit of my stomach that they are going to follow in Radio Shack's footsteps and *not* tell me much more than I already know.

From my point of view, the PET is really the third product from MOS Technology, preceded by Jolt and KIM (see Photo 1). Although the Jolt is produced by Microcomputer Associates, its debut was a result of their synergistic relationship with the then almost-unheard-of MOS Technology. It was the first microcomputer to really take advantage of read only memory (ROM) to reduce hardware.

Of course, others, such as Dataworks, with its 5K of operating PROM, preceded the Jolt (and Altair) by nearly a year—but the accent was on a firmware operating system, not a hardware trade-off. The 6530 mask-programmed chip, which combined ROM, RAM, COUNTER and I/O, was, in my opinion, almost as big a milestone in large-scale integration (LSI) progress as the microprocessor itself . . . not so much as a technological breakthrough (competing devices had similar technology), but as a practical adaptation of an emerging technology to take a giant step forward on the path of progress.

Instead of needing a single board for a Teletype port (as on my Altair 8800), the whole Jolt took up less than half the real estate. The forever-drifting adjustments of the Altair were replaced by a ROM/I-O, which measured the speed of my TTY and adapted itself! But the real value of the TIM (or Demon) 6530 was the *documentation*. Here were 1000 bytes of I/O and operating system available at power-up, and documented in such a way that its subroutines could be (and were) used in every program I wrote. It also served as a workbook for learning practical usage of the 6500 code.

The Jolt had one big disadvantage—for practical purposes: I had to have a $1000 TTY for a $300 microcomputer.

Enter the KIM

Vertical integration started with the KIM. KIM used two 6530s to double the firmware and utility. It preserved the TTY I/O of the Jolt but eliminated the total dependence on the TTY. It had its own hexadecimal keyboard, hexadecimal readout and cassette storage to replace punched tape. The single board (plus power supply) KIM outperformed three or four boards in my Altair and Imsai.

It was the end of the octal-binary (switch, LED, front panel) I/O era. Toggling data one bit at a time with lever switches was popular in the late 50s and early 60s. With data in the 3-bit *octal* format, the numerical readouts, keyboards and printers of the late 60s and early 70s became popular. Although my PROLOG preceded KIM by a couple of years in adopting hex (hexadecimal), it cost nearly ten times as much and used a 4-bit word.

The obsolete formats aren't dead, yet. Heath took (in my opinion) a step backward with Octal I/O; my new Motorola Educator II uses the single-bit format. All in all, KIM was indeed a big step forward in its time.

At the time KIM was introduced, several other I/O developments were also emerging. The highest impact devices were the full keyboard and TV display. The pioneering laurels for bringing the digital TV display out of the high-priced range (over $1000) and down to where you and I could afford it belong to Don Lancaster, who literally wrote the book on the subject. As the demand for low-cost full alphanumeric keyboards produced larger volumes, the cost came down.

> "PET gets As in three categories—vertical integration, good engineering and advanced technology."

A third development was also underway—BASIC. High-level languages (including BASIC) had been around for a long time, but, without full alphanumeric I/O, the computer hobbyist had to work on the bit, octal or hex level. This meant working *only* in machine language if you had the minimum computer configuration, such as KIM, Scamp, etc. It took only a few hours of "bit-banging" with op-code conversion to realize that there had to be something better—probably BASIC or an alphanumeric assembly language.

Even if you could afford the extras to interface the necessary keyboard and CRT (around $1000 a few years ago), there were other problems. On the hardware side, you needed memory—lots of it. You could use RAM, and wait and wait to load BASIC or an assembler. Or, you could pay and pay ($425 for my ALS-8 assembler on PROM) to get a resident assembler, BASIC or both. An even more expensive memory alternative was, and is, the floppy disk, with magnetic bubble devices warming up in the bull pen.

On the software side, BASIC has been evolving. Spurred by the San Francisco community, in general, and Tom Pittman, in particular, the old original Dartmouth BASIC was first freeze-dried to miniscule proportions and then extended. But what is more important is the cost of good software. In the late 60s, even moderate software sold for thousands of dollars per program, with additional hundreds to adapt it to your system. Contrast this with Tom Pittman's Tiny BASIC at $5, Chuck Crayne's 6800 Assembler or Ed Smith's Trace/Disassembler in the $10-20 bracket, and the stage is set for mass usage of computer power. Mask-programmable ROMs could utilize this software at reasonable prices, but only if high-volume sales could amortize mask costs.

The time has come for an affordable computer that does *not* require the fervent learning and application of hardware and software skills heretofore required of a hobbyist.

Enter the PET

The third entry from MOS Technology (a fourth is on the drawing boards) is another significant step forward for its time. At the time I paid my deposit of $800, the closest competition providing similar specs cost more than twice as much. The Radio Shack TRS-80 is squarely in competition with Commodore's PET, and the factor of vertical integration is likely to keep the field small. Only a few companies, such as Texas Instruments (with their wristwatch and calculator mass production-marketing technologies), have the high-priced chips to pay the entry fees into such a marketing race. Let's take a look at what vertical integration has done for the PET.

MOS Technology started as an independently financed splinter group from Motorola's 6800 development program, with associated legal problems (now resolved). The resultant 6502 microprocessor started as a "cheap" 6800. It uses most of the 6800 instruction set, but is (in my opinion) severely hampered by its lack of a double-byte accumulator. This deficiency is somewhat offset by page zero double-byte indexing capability, which I've never really been able to master. Others have, however, and the 6502, which seemed to come out of nowhere, burst onto the scene in the Jolt as a showstopper at the 1975 WESCON show.

I personally feel that the real innovation was the mask programming of the MOS Technology 6530 I/O chip. In any case, MOS Technology was off and running, nipping at the heels of the well-established Intel 8080 and Motorola 6800.

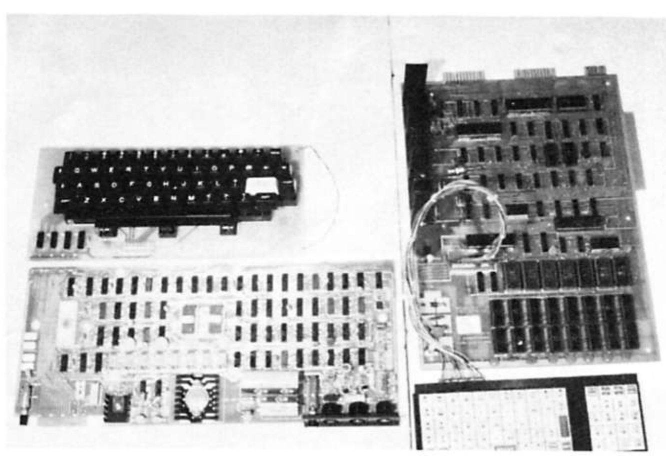


Photo 2. Front-runners: TRS-80 CPU and keyboard, left. PET's CPU and keyboard, right.

As the price of 8080s and 6800s fell under $30, the 6502 lost its price advantage, but it was staying ahead in other areas—primarily the KIM. Intel's Intellec and Motorola's Exorcisor development systems ran into thousands of dollars; KIM was less than $300. Although it didn't do nearly as much as the "biggies," KIM, with its superb documentation, was an entry into the world of microprocessors for the smaller electronics manufacturer.

Until very recently, the lack of a good, cheap assembly language and trace has limited my use of the 6502. The availability of Chuck Crayne's assembler for use on the Sphere 6800 and Processor Tech's ALS-8 for the 8080 has diverted my attention from the 6502. My biggest disappointment with my PET is the virtual nonexistence of the advertised "system monitor." It might have filled this 6502 assembler void.

Initial forays into a new field, such as microprocessing, are usually on a small scale, so the KIM filled the bill admirably (and still does). 6502s were designed into new products, and MOS Technology grew. It added memory chips to its line, which included character generators as well as the 6502 family.

The Jolt and KIM were both blockbusters when they were announced . . . but what do you do for an encore? The Apple-II and Ohio Scientific Machines had pushed the use of ROM operating systems and hardware/firmware trade-offs right up to the state of the art for 6502. Something radically different was needed.

Enter Commodore

As an early front-runner in the pocket calculator revolution, Commodore faced the same overproduction, price cutting and market-saturation problems that had left a worldwide trail of corporate corpses. Mits was almost one of these, and we all know what saved them from disaster.

The microprocessor originally evolved from calculator technology—the field in which MOS Technology also started. Today, the calculator field is headed in two directions: the $5 cheapy and the $600 wristwatch-calculator and/or the sophisticated programmable printing calculator with long-term memory. How could Commodore compete with TI and others who had vertically integrated to produce everything "in-house," from LEDs and keyboards to LSI chips? You guessed it—they bought MOS Technology. Commodore is still in the calculator business, but you have only to look at their stock-market history during the last year to see where the action is, or isn't.

When Commodore acquired MOS Technology (and Chuck Pettle), the PET was inevitable. The pieces fell into place. The major expense items for an inexpensive computer were no longer the microprocessor chips (less than $10 in quantity) nor the I/O chips, but rather the I/O devices. The TV headed the list, followed closely by the keyboard and cassette recorder. The next generation of microcomputers would require all of these . . . but was it practical?

There was the spectre of Sphere. Note the marked resemblance between the brand new PET and my two-year-old Sphere in Photo 3. The resemblance is more than skin deep. The built-in TVs and dual keyboards are obvious; not so apparent are the following: a 10K ROM operating system in the Sphere (14K for the PET); 36K RAM for Sphere (8K for PET); PIA, dual cassette, TTY and modem for Sphere (dual cassette, IEEE, PIA and TV for PET). Making allowances for cost of RAM, PROM, etc., a Sphere that was roughly equivalent to my PET would have cost about three times as much. The problem lies in the fact that Sphere Corp. *went broke* about the same time the PET was being announced.

The 4K PET was originally priced at $500, which promptly rose to $600, then to $800 for 8K (the only model delivered, so far). Even at $800, the question in my mind (particularly after shelling out the money) was, "Can Commodore really do it?" Judging from the reaction of people I spoke to and the articles I read, the consensus of opinion was that they *couldn't.*

When the promised delivery date came and went (with the same lame excuses I've heard time and again, starting with my first Altair) I, too, began to wonder. As of December 1977, Commodore was slipping even further behind in deliveries. Does this mean that they're following in Sphere's shadow? Will my PET become another Sphere-like orphan—the Edsel of personal computers? I think not, and here's why: PET gets As in three categories—vertical integration, good engineering and advanced technology. Let's see how PET measures up to competition.

Vertical integration is, perhaps, the greatest asset. The PET combines the past experience of product development (Jolt and KIM) with the LSI semiconductor design and production expertise of MOS Technology and the "offshore" subassembly production and aggressive marketing methods Commodore developed for its calculator line.

PET's competitors have equal or greater assets in one or more of these three categories, but none can match the vertical integration of Commodore-MOS Technology. Radio Shack's TRS-80 comes closest. They have the best mass sales setup in the world. They also have the only foreign supply expertise that can rival Commodore's. This is perhaps the most important prerequisite for a cost-effective end product.

The highly priced components of a computer *system* are: TV monitor, memory, CPU (central processing unit), keyboard and cassette recorder. Competition and mass production have forced the costs of CPU production down to a point where, even if you're making your own microprocessor (MOS Technology's PET, Motorola's Educator II, etc.), only small reductions in end-product pricing can be realized.

Photo 3. Now-defunct Sphere, right, is very similar to PET, left. Note the combined keyboard, TV, CPU, integral dual cassette controls and number pad. PET's cassette is built in.

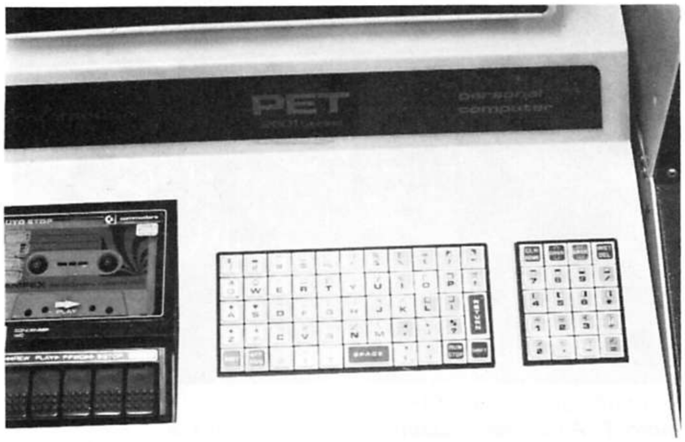


Photo 4. PET's controversial "calculator" keyboard, with quasi-standard key placement and conventional calculator number pad. Note variety of graphic symbols available with shift. Lowercase is also implemented (see text).

All the other items involve the purchase of devices and/or sub-assemblies made abroad.

The biggest item is the TV monitor. Most hobbyist computer manufacturers gloss over this item with phrases like, "Use your own television set with adapter (not supplied)." A reasonable frequency for your TV set mathematically limits the readability of characters to 16 lines, 32 characters per line, caps only.

Most hobbyists soon find that this limitation, plus competition for time on the family TV, leaves little choice but to purchase a monitor. A commercial TV monitor with adequate bandwidth for lowercase, long-line displays can cost almost as much as the computer (before it "grew"). In fact, I'm using monitors that cost as much as my PET. Inexpensive but adequate TV electronics come from Japan, Korea, etc. So does another major accessory—the cassette recorder.

The competition now takes on an international flavor, and International is Commodore's middle name. Most of Radio Shack's line of Realistic products are also imported, including the TV monitor and cassette recorder, which account for *one-third* of the cost of the TRS-80 system. PET's keyboard is also imported (more on that later).

Speaking from personal experience, I can say the business of getting production quantities of proprietary-designed high-technology hardware from overseas is a major accomplishment. Delivery and quality control require on-site monitoring, which necessitates a truly *international* organization with established operations in the Orient.

Both Commodore and Radio Shack can do this . . . but, can anyone else? This is probably the most important factor in vertical integration—it separates the men from the boys in low-cost, high-volume production. It's possible that these two leaders could produce more cheap personal computers in 1978 than all their competitors combined—and make money at it. Even with years of calculator experience, however, Commodore is having overseas production delivery problems (as of December). On the other hand, the TRS-80 is having problems getting its full BASIC underway.

PET's vertical integration includes LSI production by MOS Technology, and when the dust settles down, this may well be the deciding factor. Initially, the TRS-80 had an edge because it was designed with LSI already in high production from second-sourced suppliers. MOS Technology has had to cope with the learning-curve problems of getting their new LSI RAM and ROM chips into overseas production. These two items account for most of the costs of the respective CPU boards (see Photo 2).

As the learning curve progresses, the tables should turn and give the PET a clear-cut advantage over all comers. PET's in-house volume base for the 6550 (4K, 5 V, static RAM) could even make this chip a dark-horse contender in the 4K memory field. In fact, I'm so impressed with its performance (despite four defective chips) that I'm designing it into a 6800-based controller system.

The third factor in vertical integration is marketing. In this area, the small (often garage-type) computer company is going to have a very, very rough time in the next year or two. Radio Shack, with its massive string of franchised outlets, has a clear-cut advantage, and its parent company (Tandy) is opening a string of computer stores.

In marketing, the PET is a phenomenon, so far. The Commodore calculators survived in a cutthroat marketplace; so this, along with KIM, gives the PET a solid foundation. It's been further augmented by bringing in experienced personnel from competitors in the field. Any newcomer will think twice before going up against this kind of marketing competition—the blue chips in this game are expensive.

Another factor—the concept of utility—sets these contenders apart from their predecessors. They are not aimed at the hobbyist computer-addict market (although the impact will probably hit 7 on the Richter Scale).

The Keyboard, Graphics and "Extras"

Both the PET and TRS-80 have recognized that the family appeal requires electronic game appeal. This makes a TV and keyboard graphics mandatory and brings up the problem of keyboard and/or joystick input. Although both have graphics capability, neither has a joystick (as does the Dazzler or Apple-II). I'm sure that this will become available in the future since both have expansion capabilities to support a joystick.

There is a basic difference in the use of graphics in the PET and TRS-80. The TRS-80 splits each character block into a decoded matrix like the Cromemco Dazzler, Apple, etc. The PET goes a different route; it gives a unique graphics symbol to virtually every key on both keyboards. This provides a very large selection of fine-line picture elements not achieved through the older techniques. It also provides unique game-playing symbols, such as the card characters of hearts, clubs, spades and diamonds. Descending lowercase characters (with shift) for all alpha characters and reversing white-on-black to black-on-white are also provided.

All this flexibility poses several keyboard concept and design problems, since each letter key must display six different characters. How can it

Photo 5. My Sphere's original alpha keyboard was replaced as shown. Note pasted editing and control labels on fronts of keys. Specialized timing controls at far left are not standard.

be done economically? PET's solution was, of necessity, a compromise. By using two calculator-type keyboards (for which Commodore tooling was probably available) and changing the artwork on the anodized caps, they got an inexpensive (probably the cheapest in the world) alphanumeric keyboard. The alpha key arrangement is only quasi-standard, but the separate calculator numeric keypad is standard. It is also small enough for the cassette mechanism to be mounted alongside it and still fit a minimum-size case.

Both keysets are mounted on the same cost-effectively designed passive motherboard. Since the keyboard matrix plugs into the CPU with a single cable (see Photo 2), it would be possible to use a standard-spaced keyboard in parallel with, or instead of, the calculator board.

The most commonly criticized feature of the PET is the key placement of the keyboard. Keys are more closely spaced than normal, the middle row isn't staggered, and the feel of a calculator key isn't the same as that of a typewriter (it's more like a Teletype). I was told (by a TRS-80 booster) that it is impossible to touch-type on the PET. He was wrong; however, it does take a relearning period, much like going back to a stick shift after driving an automatic for years.

When I returned to my full keyboards on the Sphere and Imsai, I realized that I've always used the hunt-and-peck method for number pad entry, multiple key-control character and special character entry. Unlike touch-typing, most of my programming is really hunt and peck, and the PET is just about (but not quite) as easy to use as the Sphere (see Photo 5). A programmer friend and one of our keypunchers both claim that PET's keyboard drives them up a wall. But then, how many PET customers are professional data processsors?

I understand that the next model PET will have a full keyboard, but will cost a lot more. I could easily wire a $40 keyboard to replace the original—in fact, the original lousy alpha keys on the Sphere shown in Photo 5 have been replaced, just that way. Then what would I do about the 70 graphics and special characters that aren't available as standard key tops? In short, PET's keyboard isn't great, but neither are the practical alternatives.

Another quirk of the PET is that its graphics and lowercase display modes are mutually exclusive. It initializes to the graphics mode, and requires a POKE 59468,14 to convert the display to lowercase. A POKE 59468,12 returns to graphics. This is accomplished by some mysterious hardware/software manipulations involving a PIA and ROM that I haven't deciphered, yet. You can't mix lowercase and graphics. Changing modes changes *every* shifted character on the screen, but not in memory. It can create some weird effects that I used to change graphics each second in an experimental STOPWATCH program. PET didn't list its lowercase capability in specifications at the time I bought it, so it came as a pleasant surprise—one of several "extras."

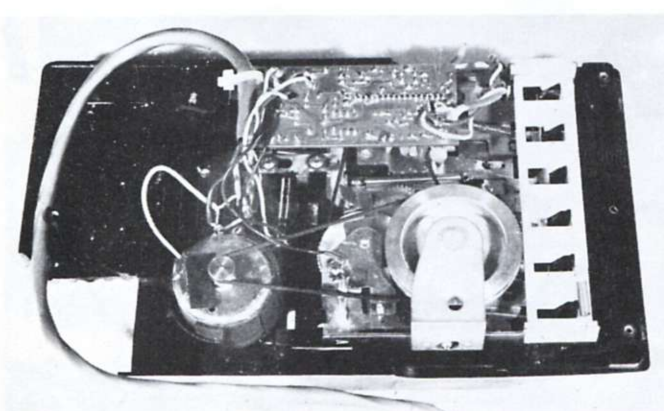


Photo 6. Gutted cassette is probably a stopgap measure. Note absence of usual electronics, speaker, jacks, etc.

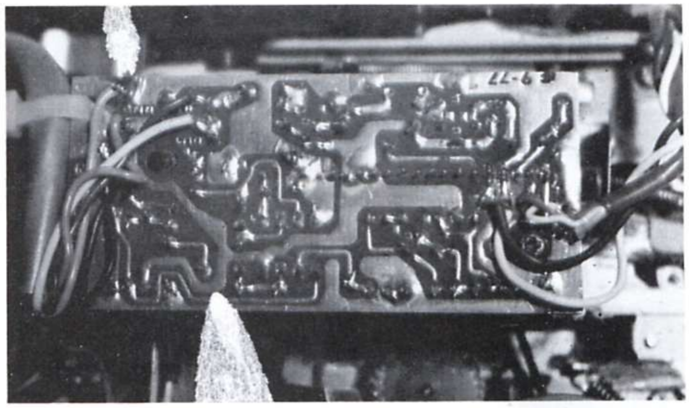


Photo 7. All of the cassette record, playback and erase electronics are on this single small PC card.

A real-time clock is another of these extras. It doesn't do as much as an S-100 real-time card, but it doesn't cost an extra $130, either. It outputs a six-digit, 24-hour clock word, e.g., TI$ = 235959 = 23 hours, 59 minutes and 59 seconds. At 240000 it resets to 000000 and is software presettable. It also outputs JIFFIES, which are 1/60 second counts accumulated from 000000. JIFFIES are about as fast as anything you could use with any program written in BASIC. The clock runs off of the 8 MHz crystal.

Although it isn't immediately evident, the real-clock function is an excellent example of the aggressive design policy that makes the PET a technical step forward, regardless of price. I haven't figured out exactly how they did it . . . but what I've deciphered so far indicates an impressive utilization of the latest LSI capabilities from MOS Technology (more vertical integration here, and a valuable feature not available from their competitors). Among other things, the TI (time) function is a fundamental building block in automated home programming. Since it runs on interrupts, it will keep the time of day as long as power is left on. But, unless you trim the oscillator, you'll have to keep readjusting the readout.

The Recorder System

By now it should be evident that the PET's low price was not achieved by making a cut-down, stripped version of older technologies. Take the built-in cassette recorder, for instance (Photo 6). In all my other systems, built-in recorders are not provided. Except for the TRS-80 and PET, cassette recorders are a "hidden" extra expense of personal computing. The garden-variety cassette recorder isn't optimized for digital recording. It sacrifices signal-to-noise for low harmonic distortion and ignores phase distortion. Its electronics are an

"Except for the TRS-80 and PET, cassette recorders are a hidden extra expense of personal computing."

overkill, including automatic gain control which prevents full-level recording.

PET's cassette takes a radical departure. All the erase-record-play electronics are on the single card shown in Photos 6 and 7. Obviously, the gutted mechanism in the current models is a stopgap solution to overseas delivery problems, and the eventual recorder should be produced at a significant savings over competing systems.

The recording method is a compromise between dc saturation digital recording and the frequency-shift-audio techniques currently in vogue. Dc erase is used, and square waves are fed directly to the record head. The record current is limited to prevent complete saturation and biased for centering. On my unit this results in about 8 db better signal output on playback with improved phase distortion characteristics. My unit also had two dry-joint solder intermittents.

To find these, I had to create a schematic. I also needed the information to find out why my PET played back its own tapes flawlessly, but couldn't copy from one cassette to another as I've been doing with my Sphere, etc. The problem was in the reduced record level and phase distortion. It worked most of the time, and might even be practical for short programs, but it certainly isn't good enough for longer ones or file storage. PET got some demerits when I found that several playback errors were *not* caught by the double-recording check. I'm sure that a mass cassette duplicating operation will eventually duplicate digital tapes in this format, but my copy of the first one on the market (*not* Commodore) was a disaster.

I asked Chuck Pettle if PET was designed that way on purpose to give Commodore an edge in the prepackaged software field. He was surprised at my difficulties, and assured me that the intention was to provide a truly interchangeable format for all PET users. There is no problem in interchange of original recordings, only duplicated copies.

I really notice the absence of a counter on the cassette recorder. Unless you restrict your tapes to two or three per side, you wait forever for the playback to find the right program. In desperation, I use a separate recorder to find the approximate start position with a counter and then transfer it to the PET—a real pain. Although the baud rate is high (1100 baud), a long preamble, double-buffered recording scheme and a motor stop between files slow down the file handling to a snail's pace, compared to a Tarbell. The second cassette port is fully implemented on the CPU but, as yet, no recorder is available to make use of it. I hope it will have a counter.

Another nice added "extra" is the *verify* mode. After recording, you can rewind and verify the tape playback against memory. Since I've eliminated the intermittents in the recorder, it's a bit redundant because there has *never* been a playback error.

Another extra is the unrivaled simplicity of loading a program—turn on power, insert a cassette and press RUN. It tells you to play the recorder, displays the label of the first thing it finds, tells you it's loading and if it loaded OK and runs the program. Even a very small child can do it. An A rating. If children are to realize the maximum educational potential of personal computing, this approach will be very helpful. If you specify a label, it will display each label it finds until it gets the right one—then loads it.

The PET's recording format is unique, like those of most of the new computers . . . it looks as though the Kansas City standard will bite the dust. The PET maximizes the hardware/software trade-off. It uses almost a bare minimum of analog devices (room for design improvement here), a couple of PIA ports and no UARTs or other serial I/Os. It's the most cost-effective digital recording system I've analyzed, although the Educator II is a close second. It's an A+ example of saving money with design ingenuity.

The TV Board

All of the other competing computers with CRTs use off-the-shelf monitors or TV modifications. In this instance, as with the recorder, PET breaks with tradition, gaining improved cost/performance by replacing hardware with firmware. The complex sync signals, which use up hardware in both the traditional character generators and monitor, are generated by firmware and the very powerful 6522 I/O chip. The video, horizontal and vertical drives are also available on the rear user terminal. Because the video board doesn't need to decode sync or amplify video, it's simpler (and cheaper) than competing models. Since the screen is built in close to the operator's eyes, it can be smaller than a separate monitor (such as the TRS-80) and still provide the same legibility —another saving.

There is only one external adjustment: contrast. My PET needed vertical centering. It was done with the black tabs on the neck of the CRT. A small pot at the rear adjusts the height. So far it has been very stable and provides a steady picture with a superior bandwidth, another A-rated example of cutting costs with creative system design.

The CPU Board

Photo 8 shows the CPU board—PET's brain. It takes less than two minutes to remove it. Wiring harnesses cost money. Both the PET and the TRS-80 keep them to a minimum. The board plug-connects to the power supply, keyboard, video and recorder. Incidentally, be careful with the keyboard plug. Mine became intermittent after its first replacement. The leaf spring contacts in the female cord connector are easily overstressed and

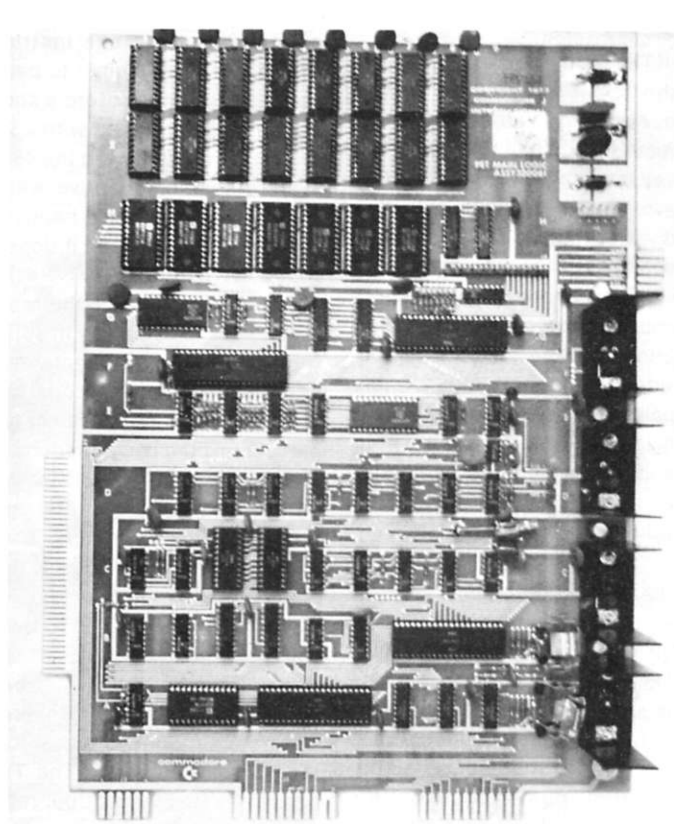


Photo 8. PET's brain: Top 16 chips are RAM. Seven ROM chips below contain operating firmware. Power supply and cassette #1 are along right side. Output ports are along bottom (rear). BUS and memory expansion are at left.

may have to be re-formed with a probe.

Note how the four expansion connectors are made directly to the board through slots in the side and rear of the case—a far more efficient arrangement than that of any of my other systems. At this time there is nothing available to connect to them, but when there is, the difference between the utility of the PET and the TRS-80 is likely to give the PET a big competitive edge (see Photo 2).

The long connector on the left-hand side has what the TRS-80 has on its single expansion port. In addition, the addressing is available decoded into 4K blocks. Current plans call for its use in RAM, ROM and PROM (2716) expansion. The monitor and assembly language will probably go into ROM.

The current price of $200 for 4K of RAM makes PET about the highest-priced RAM on the market. When the 6550 moves out on its learning curve, PET should be in a position to provide the cheapest memory around.

The small connector pad in the lower left corner is for cassette #2. You can play the recorder into it. It works, but, as yet, there isn't any recorder available to use with it. If PET doesn't make one available soon, I'm sure someone else will, and I hope they provide a counter. The center connector brings out the aforementioned video feeds and half of the powerful 6522 PIA programmable I/O. It's called a User port and, if documented adequately, could become PET's most valuable asset.

The lower right connector is the IEEE-488 bus. If and/or when the S-100 bus system yields to another format, it's likely to be the 488. This system is supposed to allow your PET to talk with up to 18 peripherals through a high-speed, 8-bit parallel bus. Properly implemented, it can be almost as fast as a motherboard or backplane.

There are more than 200 devices (a lot from Hewlett-Packard) available for use with the 488. However, most of them cost more than the PET and are special-purpose test instruments, not really suited to personal computing. Motorola and others are coming out with LSI chips that should make the 488 system cost competitive with the S-100. This won't happen immediately, but when it does, PET will have a well-established lead over the rest of the pack, particularly in software. PET gets an A here because Commodore's vertical integration should allow them to make inexpensive peripherals that could be used with competitors' microcomputers, as well as with the PET. The TRS-80 (see Photo 2) with its single, unique 40-pin port only rates a D when it comes to this kind of expansion.

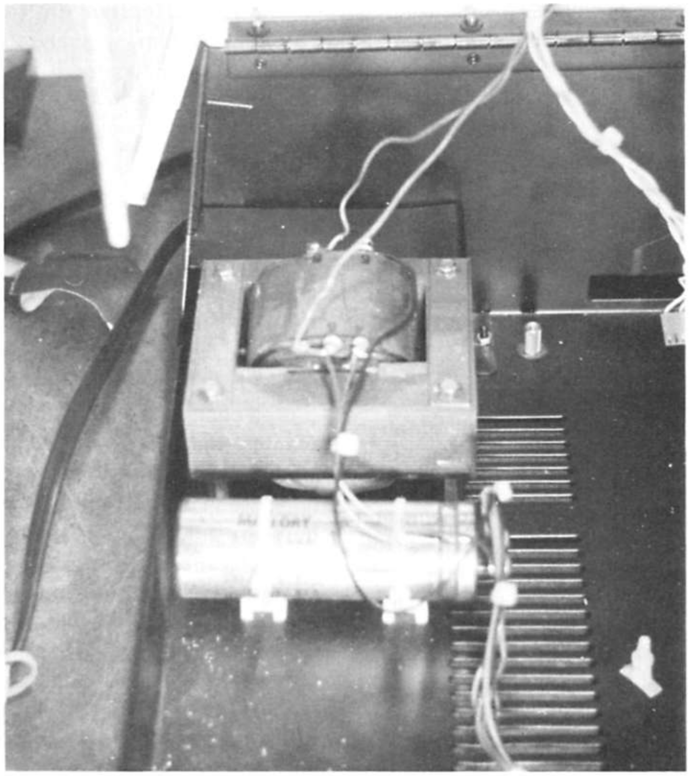


Photo 9. The power transformer, filter capacitor and 110 ac control are the only electrical devices directly wired to the chassis.

PET's power supply, see Photo 9, is 5 volts *only* (Sphere uses five different voltages) for the digital equipment. The TV board has its own rectifier-regulators. The CPU board splits the load into three sections with the three 5-volt regulators along the left-hand side. The two power transistors with heat sinks are the motor controllers for the cassette recorders. The regulators are running hot now, so additional loads should be limited.

The 8 MHz crystal clock drives the 6502 microprocessor at 1 MHz. It also provides the TV timing and 60 Hz JIFFIES. The crystal is stable, but the factory feels that plus or minus 1½ minute per day is adequate. If you want greater accuracy, you'll have to trim the driving capacitors next to the crystal. A 6-30 pf variable in parallel with 22 pf did the job for me (see Photo 10). Now I can trim it like my digital wristwatch. The 24-hour clock is counted with interrupts and should be software independent. I've encountered unresolved problems with a program that continuously reads TI$—it speeds up the displayed time.

The 6550 RAMs are 4K, high speed, low power, static, and require only 5 volts. They are pinned as 1K by 4 bits, so they are socket-mounted in pairs along the front of the board. Page 0 is at the left and the high nibble is toward the front. If memory problems occur (I've had four failures), you'll need to play musical chairs, since it's impractical to apply a memory test to the low 1K where BASIC operates its scratchpad. This device gets an A for design and a D for deportment.

The ROMs in the first units (mine included) were not the MOS Technology devices currently being shipped. They are 2K devices and are now being soldered in. Although PET is officially specified for 14K of ROM, 2K of the same ROM is used as a character generator. The PET is currently oriented toward the personal-computer mass market; changing *only* the ROMs and keyboard caps could make it a super development system, smart terminal, dedicated controller, word processor, typesetter or just about anything micros are, or will be, used for. It could happen virtually overnight, and, with the inherent mass-production economics, it would be a price-cutter in any market. (That's awesome when you think about it, since MOS Technology could supply inexpensive masked ROM for any application.)

Mechanical Engineering

PET gets a B+ for its metal case. It will probably be replaced by a more durable plastic case, but dies for this size molding are a long time coming. In either case, the PET is utilitarian and its exterior appearance can only be compared to units costing several times more. It even has a prop to hold up the hinged top for servicing. The tooling is a little sloppy and some of the holes are mismatched. My degree was in mechanical engineering (a long time ago), and I appreciate good mechanical design. PET has it. Not only is the case impressive, but so are the circuit-board layout and the overall cost-effective design decisions. Three of the four circuit boards are inexpensive "single sided."

The case of the TRS-80 is a good design job also, but the overall effect looks like a keyboard with dangling wires to a dominating TV, with a cassette and power supply strung around it. The TRS-80 is more attractive than the uncased Jolt or KIM, but, to the average neophyte, it may not *look* like a computer when compared to a PET or Sphere.

And Now . . .
The Bad News

PET gets low marks in two areas: reliability and service maintenance. I give it a D. At the same time, there is enough room for improvement so that it could go to the head of the class. It worked when I received it. Since then, I've had four intermittents; three were bad solder joints and the fourth was a defective connector. I have also had four memory failures, a glitch in my TV horizontal sweep, drifting vertical centering, undetected read errors, off-frequency crystal calibration and a couple of other weird goings-on that remain unidentified. To put things into perspective, I should add that this behavior is *better* than that of my Mits 8800, Mits 680, Imsai, Sphere, Jolt or SWTP.

Bugs are a way of life when you get the first units off a production line; I expect them. Mits had trouble with bad memory chips on the first 8800 boards . . . worse than my PET's. They wouldn't send replacement ICs so I reluctantly sent the useless boards back. It was four months and $40 extra before I got working memories from them.

The big hang-up with bugs in my PET is that there is no service information provided; furthermore, it's unlikely that I'll see a schematic for a long, long time, if ever. The local distributor doesn't have any more information or spare parts than I do. The 6550s aren't on the market and there are no complete spec sheets available for them. A magazine article had estimated that factory service would require two months, including shipping. If you detect a note of frustration, you're right! It's even worse when you see a little LED on the board and know that it's a part of a built-in diagnostic system that's using up some of the ROM you bought. Neither you nor your local dealer can use it; it's a factory secret. Now what do you do?

First, call the factory. When I called, the girl who answered didn't know what I was talking about, and the fellow who might have known was unavailable. People who went through this with Mits and SWTP in the "old" days (it's changed now) know the script.

After a period of fuming and fretting, punctuated with expletives, I decided that $10,000 worth of test equipment and four years' worth of experience with microprocessors ought to be able to solve the problem without schematics. It did —partially.

I had to write my own memory test program and use a multitrace storage scope to eventually find the intermittents and some of the bad memory chips (also intermittent). Then, another call to the factory. This time I was put through to the right man with the right attitude and right answers—a real gem. Three days later I had replacements and spares, no extra charges, no insistence that I relinquish my cherished PET for an indefinite stay and a lot of good solid advice on how to tackle the remaining problems. He also assured me, as did Chuck Pettle, that most warranty repairs took less than a week, if worse came to worse.

OK, so my PET is running pretty well, but what about the housewife in some boondocks town without a well-equipped laboratory, years of experience or a WATS line? What if she got my #171? Well, as of December, her only recourse would have been to return it to California or Pennsylvania and hope that Murphy's Law, as applied to intermittents, wouldn't require too many return trips. However, by the time you read this, PET could be in the best service position any personal computer manufacturer has ever been in.

The information and special wiring harness should be released so that the built-in diagnostics can be utilized by relatively inexperienced people. Faults could be fixed by identifying and exchanging the offending circuit. Since there are only four circuit boards and a rudimentary power supply, the built-in diagnostics, augmented by test cassettes, should easily bracket the problem.

From personal experience, I'd estimate that most "while-u-wait-repairs" could be done in less than 15 minutes. The ability to do this was obviously a design objective. Currently, there are two flaws in the grand plan. All available parts are being used to try to satisfy a huge backlog of delinquent system orders. There are no spare boards for dealers or servicemen. Also, documentation and test equipment are not yet available in what Chuck Pettle describes as an "acceptable" form.

When I asked him when I would get schematics adequate for servicing the problems with my PET, he told me that only the characteristics of the I/O were going to be released, and the rest would be kept "secret from competitors." In a vain attempt to get him to change his mind, I pointed out that a competent computer engineer could produce a schematic of the whole system in a few days and that any programmer who has written a BASIC interpreter (see "A Tale of Four BASICs," *Kilobaud* No. 13, January 1978) could produce a source listing of the ROMs. In fact, the only firms that possess these in-house skills *are his competitors*! As they say about gun control, "If you make gun possession a crime, then only criminals will possess guns." If PET (or Radio Shack) refuses to supply schematics to servicemen and product designers, then the only people who can get the information are their competitors with skilled manpower.

I admit I'm biased by the many wasted hours I've spent debugging my PET, but I can't help feeling that Chuck is adhering to a shortsighted policy. However, I feel that he's a reasonable man, so I hope someone else will succeed where I failed, and we'll all benefit.

After rereading what I've just written, it's evident that, with the exception of the service and documentation problems (which may not exist by the time this is printed), the PET has been depicted rather positively. As a matter of fact, Commodore could easily drop a perhaps fatal wad on the PET venture. Several local dealers who

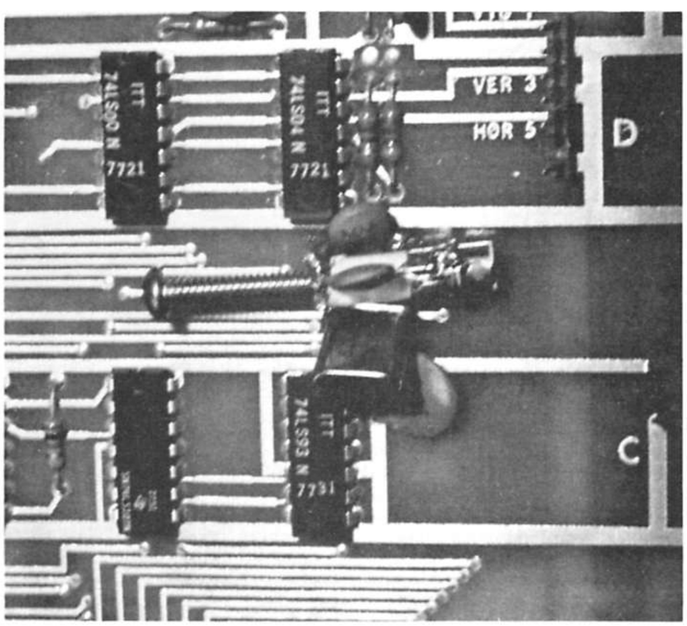


Photo 10. Author's modification of 8 MHz crystal oscillator with trimmer capacitor trims 24-hour clock to high precision, but software problems remain.

were pushing PET a month ago are now telling customers to buy something else because "Commodore is going broke." I suspect that delinquent deliveries and "cash-up-front" dealer policies are the real motivation, but how much of this can PET take?

One look at the gutted cassette recorder implies a big problem with overseas supplies. Less obvious, but unmistakable, evidences abound to attest to the probability that my cold-solder-joint intermittents are the result of questionable production practices and relaxed, or inadequate, quality control.

No matter how cost-effective a product design may be or how dynamic the pre-production sales effort, if you can't produce a reliable product on schedule with efficient and minimal after-sales service, you'll lose the ball game . . . remember Viatron? MOS Technology had problems with the early KIMs (mine went back twice), and successfully solved them. I'm betting that PET will have a similar success.

When it comes to software, PET gets a C, with an "incomplete" noted in the margin. The bare-bones listing of Microsoft's latest BASIC makes it difficult to work with, much less evaluate. Someone else will have to do that after the manual is published. So far it's about the same as the Crayne BASIC I've been using on the Sphere and the Mits on the Altair. It's faster, the error messages are better and the files are double buffered, but watch out for commas within quotation marks, such as addresses in FILE programs—they tend to act as delimiters. If you're used to using abbreviated instruction, you'll be disappointed.

> "If PET refuses to supply schematics to servicemen and designers, only competitors can get the information."

The original specs called for a 4K basic operating system. Compared to my Sphere operating system with only 2K of PROM, the PET is a disappointment. There are USR and SYS commands in BASIC, but no facility to load or generate machine code except by writing your own program to POKE it in BASIC. I had hoped that they would at least start where the two-year-old Sphere system left off.

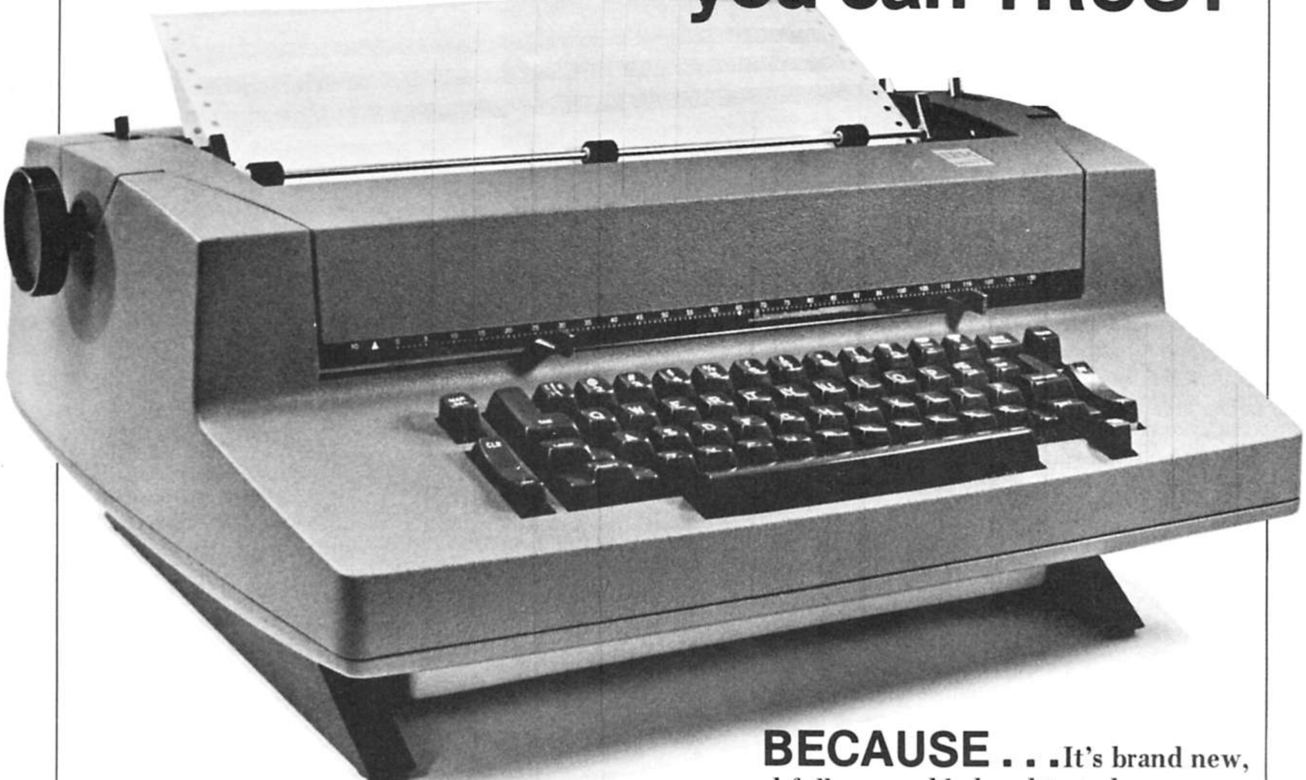
If I were to put the Crayne's Sphere BASIC in ROM along with the current ROM operating system that consists of V3D, PDS, Mason's X-DBUG and Programma Assoc. text editor, it would require 20 percent less ROM and provide many features not found in this version of the PET. This includes utility subroutines such a number-base conversion, multibyte division and multiplication, block moves, hex-decimal-ASCII conversions, etc.

Conclusion

After all the pros and cons have been considered, it looks to me as though Commodore's PET has the brightest future of any microcomputer I've ever evaluated. It could graduate *summa cum laude*. Right now it's on shaky ground and could conceivably *flunk out*, as did the Sphere. It could have the short-term success of the *average* microcomputer, such as the Jolt. No matter how history marks its final report card, a new era of *mass* usage of artificial intelligence has been ushered in by Commodore's PET. ■

New! **SELECTERM** SYSTEM 9710

The SELECTRIC II* Printer you can TRUST



BECAUSE . . . It's brand new, and fully assembled and tested.

BECAUSE . . . After extensive engineering design and testing by Micro Computer Devices, IBM Corporation has approved the SELECTERM for use with your microcomputer, and provides you with their factory warranty and yearly service agreement for the typewriter. In addition, the electronics conversion portion is fully warranteed by Micro Computer Devices.

BECAUSE . . . You can connect the SELECTERM to your computer within minutes of taking it out of the carton.

IT'S THAT EASY! AND THAT RELIABLE!

FEATURES
- Complete ASCII character set in supplied element.
- Full upper, lower case alphanumeric characters.
- Tab Command, Index (vertical tab), Backspace, Bell—all under computer control.
- Parallel Interface, standard.

ALL ELECTRONICS INCLUDED
- Power supply, electronics and cable sets included to permit immediate connection to the parallel port of any computer, at standard TTL level.

SOFTWARE
- All necessary conversion software in PROM to handle ASCII input, directly.

PRINTER or TYPEWRITER
- May be used as a standard typewriter when not in use with your computer.

OPTIONS
- Dual Pitch, $125
- Correction Feature, $125
- Tractor Feed Platen, $250
- Noise Reduction Feature, $50

AVAILABLE SOON
- RS-232 Interface

PRICE and DELIVERY
- Assembled and tested, $1750
- Available ONLY from authorized dealers.
- Delivery 1 to 2 weeks from receipt of order.
- OEM delivery in quantity within 30 days.

SELECTERM

micro computer devices inc.
960 E. Orangethorpe, Bldg. F
Anaheim, California 92801
Telephone (714) 992-2270

*Registered trademark of IBM Corporation

M30

"Innovators to the Microcomputer Industry"

Bob Buckman
3954 Hillview
Santa Maria CA 93454

Scope Power!

a review of Tektronix's Model 922

Tektronix's motto is "Committed to Excellence." That motto, and the resulting equipment, prices most of Tektronix's products out of the hobbyist range. In 1976, Tektronix announced the T900 series of oscilloscopes. Finally, a Tek scope I could afford! Last June I purchased a medium-priced T922.

You software types should know what an oscilloscope is. It can display little squiggly lines to enable you, or your friendly neighborhood service center, to troubleshoot your hardware. Read on for a short course in scopes.

Tektronix has always been the Cadillac of the industry, with appropriate prices. The great thing about the 900 series is that the basic design is the same as for the top-of-the-line scopes. Why should you buy this scope? What features put the 900 series above all the others in the market?

First, the scope tube itself. Designed and constructed by Tektronix, it contains no manufacturing shortcuts. The scope is literally built around the scope tube. The huge 8 x 10 centimeter square screen with internal graticule on the front of the display won't rotate out of place, rub off or fade with age. With the nominal 12,400 volts dc acceleration potential, the dot size is small and the writing speed fast.

The least expensive scope in the 900 line is a single-trace instrument that sells for $650. All the others in the line are the dual-trace variety (and only slightly more expensive).

The scope I bought is the T922—a dual-trace, 15 MHz bandwidth, portable instrument that tips the scales at a mere 15 pounds (see Photo 1). I ordered it from stock by telephone from my local Tektronix Service Center and mailed a check; two weeks later, UPS left it on my doorstep. Ah! Nothing beats the joy of opening a box with Tektronix printed on the side! It was packed securely with air space all around and contained a manual and two $42 list-price X10 attenuation probes (included in the price—$850 FOB Beaverton OR—you thought I wasn't going to tell you). It worked perfectly; that's one advantage of buying an assembled instrument.

I was surprised at the length—almost 19 inches, most of which is scope tube . . . I couldn't resist looking inside. The plastic case comes apart after removal of six bolts (see Photo 2). Most of the acton occurs on two single-sided circuit boards. The pilot light is a neon NE-2 with a light pipe guiding the way to the front panel. Most knobs are extended with plastic rods to controls positioned toward the rear of the circuit boards. The attenuators, since they are of unique stripline design, are mounted on the front panel near the BNC input connectors—altogether an easily assembled, well-planned layout, with room for expansion (see Photo 3). I may add the T935 delayed sweep features as soon as I get the other scope manual.

The 15-position calibrated attenuators are constructed using the same stripline camswitch techniques from the 500 MHz mainframe machines. With steps from 2 millivolts to 10 volts and a variable control over a 2.5 to 1 range, any voltage can be easily displayed.

The 912 and 922 calibrated time base has 20 steps in a 1-2-5 sequence from 0.5 second to 0.2 microsecond per centimeter. With a variable control from 1X to 10X, the maximum sweep rate is 20 nanoseconds per centimeter!

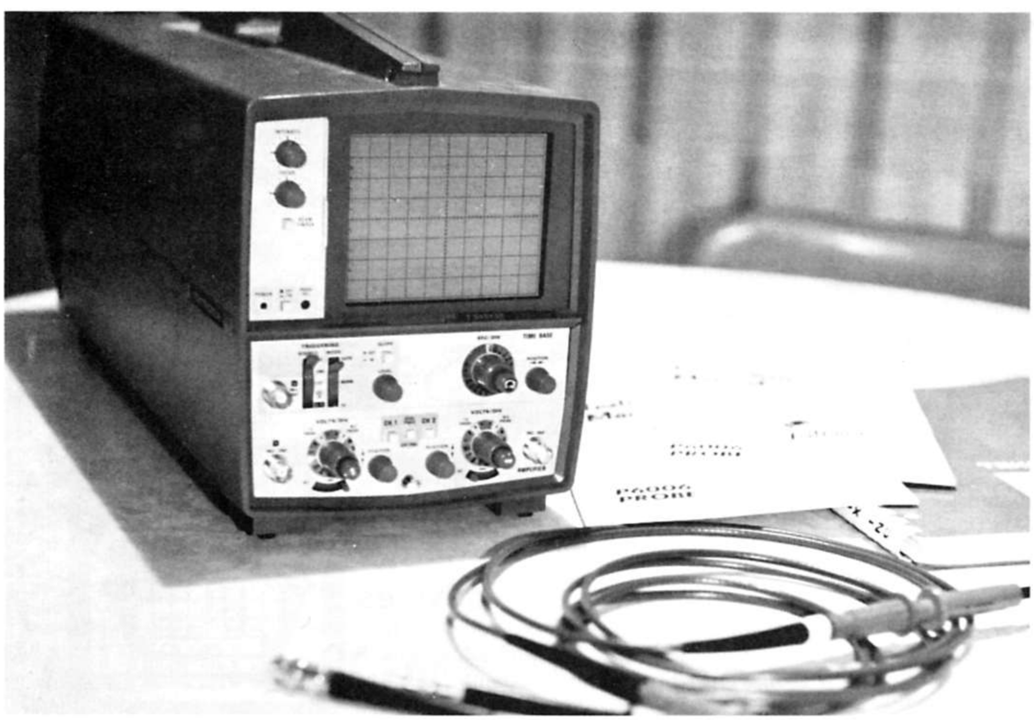


Photo 1. The complete system.

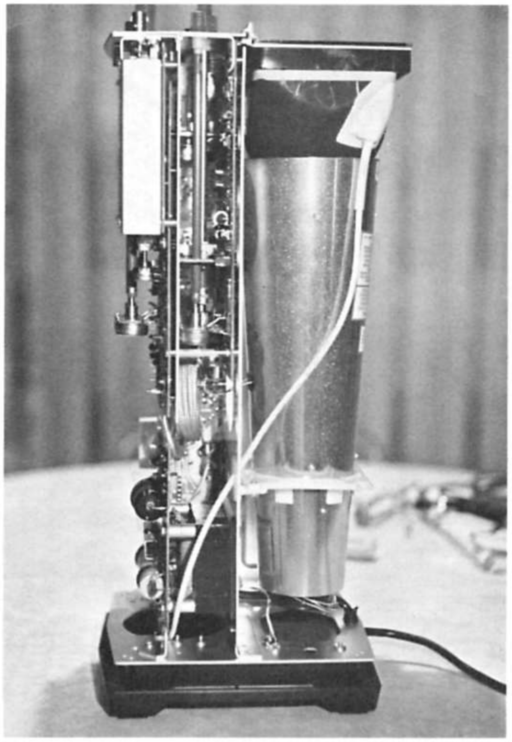


Photo 2. The tube determines the scope's length, width and height (right side view from front).

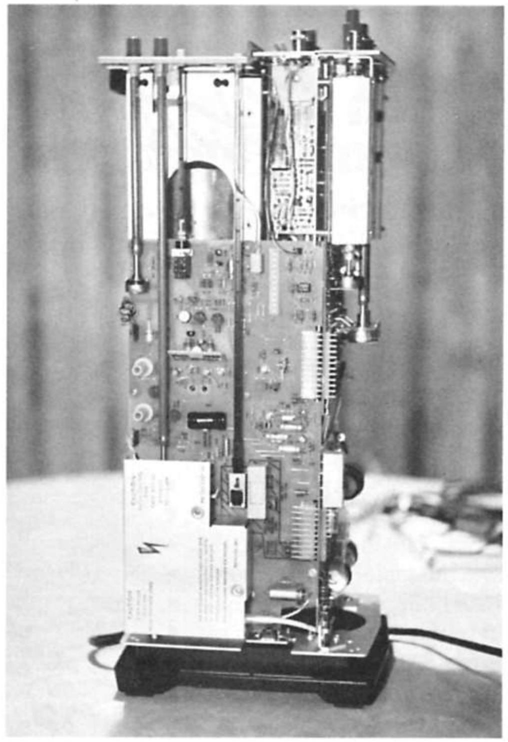


Photo 3. Most controls are extended (left side from front).

Frequency response? Oh yeah—dc to 15 MHz for the 912 and 922 (my scope), dc to 35 MHz for the 930 series. That's minimum, folks! The top trace in Photo 4 is an 8080 phase 1 clock; the bottom trace is phase 2. The slight ringing on the low side of the phase 2 clock is due to a bad ground to the scope probe (see Photo 4).

Triggered sweep assures that the sweep does not start until the triggering conditions are met. This allows you to easily measure single or repetitive pulses, which will always show up at the same place on the screen. You can select the positive or negative slope of the waveform and, by varying the trigger level, trigger anyplace on the waveform. Auto triggering, alternate or chop mode (dependent on sweep speed), TV field or line sync, external sweep and X-Y modes give you any combination of triggering modes you need.

Ever lose the trace? You know the signal is there but the dc level has moved it off the screen somewhere. Simply depress the beam-finder button. The display will be squeezed vertically and horizontally so it will fit on the screen no matter where the position controls are set. Once you've found the trace, set the controls for best position and release the momentary beam-finder button.

The balanced delay line (the looped cable in the center of Photo 2) slows down the input signal so the sweep starts before the display—you can see the part of the signal that started the sweep. No more guessing what the leading edge looks like!

The manual provided with the scope is a work of art. It's actually a scope textbook all by itself: 20 pages of operating instructions, five pages of performance tests to let you know if the scope still meets its specifications, 31 pages of service information to help you fix whatever might be wrong, at least 50 pages of fold-out block diagrams and detailed schematics, complete exploded views of all mechanical parts with a detailed parts list for everything—truly a joy to peruse.

Did I say anything about resale value and reputation? Ten years from now, I could sell this scope for nearly the same price I paid for it. Tektronix's worldwide reputation is simply the standard of excellence.

This report has been flattering. What didn't I like? Only two things bothered me. First, the chop and alternate sweep mode is selected internally by the sweep-speed switch in the T922, but the rack-mount version of the same scope has a front-panel switch! I know that the chop mode isn't very useful above half a millisecond per centimeter, but sometimes it is essential for single-shot events and I can't get to it. Frustrating. The second thing is a feature you don't really think about until it is time to take a scope photo: a scale illumination. It would require front-panel redesign and add at least a hundred bucks to the price, so I accept the lack of scale lighting. The rack-mount version does have scale illumination and takes any 7000 series scope camera—it also costs more.

If you're deciding on a scope, consider the Tektronix 900 series. Why not the best? I know my scope is one of my best investments. A card to Tektronix, PO Box 500, Beaverton OR 97077, or a call to your local Tektronix representative will result in your getting ordering information.■

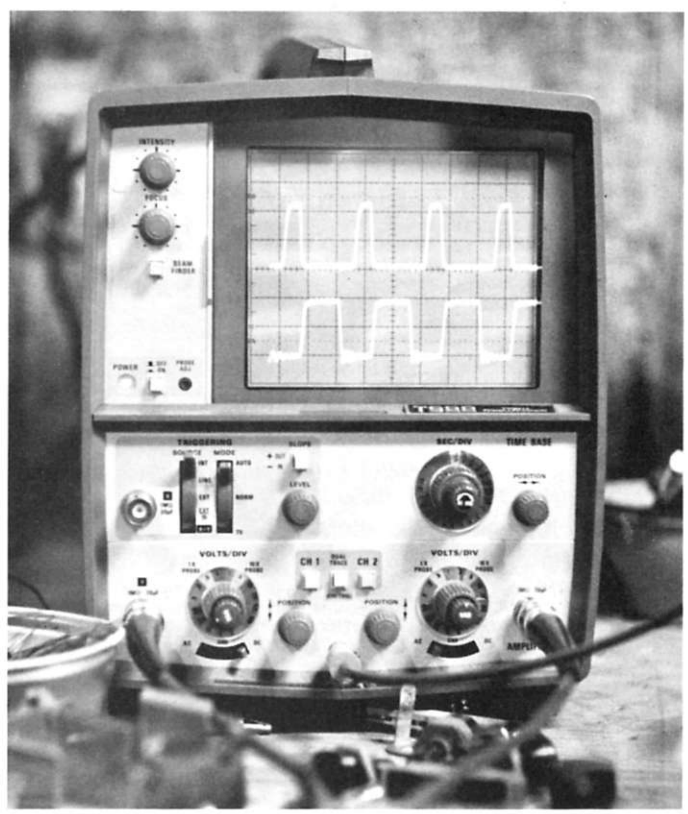


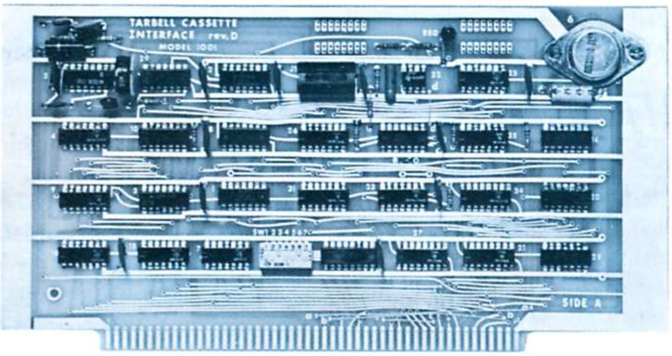
Photo 4. 8080 clocks 200 nanoseconds per division.

SEE YOUR LOCAL Tarbell Electronics DEALER TODAY . . .

ALABAMA	
Huntsville	Computer Land
ARIZONA	
Phoenix	Bits & Bytes
Tempe	Byte Shop
Yuma	Ozymandias Systems
CALIFORNIA	
Santa Ana	Advanced Microcomputer Products
Costa Mesa	Algorithm Personal Computers
Hawthorne	Applied Process Laboratories
San Rafael	Aaron Enterprises
Long Beach	Avido Electronics
Fullerton	Bits N Bytes
Berkeley	Byte Shop
Burbank	Byte Shop
Fresno	Byte Shop
Haywood	Byte Shop
Lawndale	Byte Shop
Palo Alto	Byte Shop
Pasadena	Byte Shop
Placentia	Byte
Santa Barbara	Byte Shop
Santa Clara	Bytel Shop
San Diego	Byte Shop
San Jose	Byte
San Jose	Byte Shop
San Rafael	Byte Shop
Tarzana	Byte Shop
Walnut Creek	Byte Shop
Westminster	Byte Shop
Los Angeles	Richard Chew
Costa Mesa	Computer Center
Van Nuys	Computer Components, Inc.
San Diego	Computerland
San Leandro	Computerland Corp.
El Cajon	Computer Metrics, Inc.
Westminster	Computer Playground
Lawndale	Computer Stop
San Francisco	Computer Store of San Francisco
Hayward	Computer Systems Unlimited
Indio	Dean's Music City
Reseda	Dynatron International Company
Fresno	Electric Brain
Riverside	Electronic Supply
Torrance	Futra Company
Lawndale	Jade Computer Products
Santa Ana	L. Electronics
Anaheim	Micro Computer Center
Santa Monica	Mission Control
Sherman Oaks	Peoples Computer Shop
Redondo Beach	Personal Computer Center
Torrance	Randal Data Systems
Bakersfield	R & H Electronics
Scotts Valley	Success Systems
San Gabriel	Sunny Sounds
Carson	Sunshine Computer Co.
Tarzana	Tech-Mart
San Diego	The Computer Center
Orange	The Computer Mart
Santa Monica	The Computer Store
West Lake Village	Vector Graphics, Inc.
COLORADO	
Englewood	Byte
Boulder	Byte Shop
Denver	Computer Technology
DELAWARE	
Newark	Delaware Microsystems
FLORIDA	
Leesburg	Delta Electronics
Tallahassee	Florida State University
GEORGIA	
N.E. Atlanta	Byte Shop
Atlanta	The Computer System Center
HAWAII	
Honolulu	Mahalo Microsystems, LTD
ILLINOIS	
St. Posen	Bits & Bytes Computer Store
Champaign	Champaign Computer Company
Harwood Heights	DMA, Incorporated
Evanston	Itty Bitty Machine Company
Skokie	Lillipute Computer Mart
Lombard	Midwest Microcomputers, Incorporated
Oak Park	The Computer Store
INDIANA	
South Bend	Audio Specialists
Bloomington	Data Domain
Indianapolis	The Data Group, Incorporated
Indianapolis	The Home Computer Center
KANSAS	
Mission	Computer Center - Byte Shop No. 61
Wichita	Computer System Design
KENTUCKY	
Louisville	Cybertronics
Louisville	Data Domain
LOUISIANA	
Downsville	Bill Gulledge
MARYLAND	
Towson	Computers, Etc.
Rockville	The Computer Workshop
MASSACHUSETTS	
Boston	% American Used Computer Corporation
Waltham	The Computer Mart, Incorporated
MICHIGAN	
Berrien Springs	The Abacus
Royal Oak	Computer Mart, Incorporated
Livonia	GAW Computertronics
Grand Rapids	Jepsan
Brighton	The General Computer Company
MINNESOTA	
Edina	Computer Depot, Incorporated
NEBRASKA	
Omaha	Omaha Computer Store
NEW JERSEY	
Trenton	Ace Electric
Succasunna	Computer Hut
Iselin	Computer Mart of New Jersey, Inc.
Ramsey	Dollar Planning, Incorporated
Hoboken	Hoboken Computer Works
Clark	S-100, Incorporated
NEW YORK	
Levitown	Byte Shop - East
Fayetteville	Computer Enterprises
New York	Computer Mart of New York, Inc.
Manhasset	Computer Microsystem
Hollis	Synchro-Sound Enterprises
Rochester	The Memory Merchants
NORTH CAROLINA	
Boone	Alpha Digital Systems, Incorporated
Raleigh	Byte Shop
Ashville	Computer Sharing Incorporated
Kinston	Professional Computer Associates
OHIO	
Cincinnati	Cincinnati Computer Store
Kent	Ohio Microsystems
Cleveland	Tec Mar, Incorporated
OREGON	
Beaverton	Byte Shop - Beaverton
Coburg	Forethought Products
Aloha	Kent's Komputers
Eugene	The Real Oregon Computer Co., Inc.
PENNSYLVANIA	
King of Prussia	Computer Mart of Pennsylvania
Huntingdon Valley	Marketline Systems
Pittsburgh	The Electronics Place
SOUTH CAROLINA	
Columbia	The Byte Shop No. 32
TEXAS	
Dallas	Altair Computer Center of North Texas
Houston	Andy Electronics Company, Inc.
Austin	Austin Microproducts
Austin	Balcones Computer Corporation
Houston	Computerland
El Paso	Computer Terminal
Houston	Electronic Specialty Company
Houston	Houston Computer Mart
Corpus Christi	Microsystem Services, Inc.
Forth Worth	Tandy Computers
Richardson	The MicroStore
Houston	The Computer Store
College Station	Young Electronic Service
TENNESSEE	
Nashville	Computer World
UTAH	
Provo	Computers & Stuff of Utah
VIRGINIA	
McLean	Computer Systems Store
Newport News	The Home Computer Center
WASHINGTON	
Seattle	The Retail Computer Store
WEST VIRGINIA	
Morgantown	The Computer Corner Incorporated
WISCONSIN	
Sheboygan Falls	DMA Incorporated

TARBELL SETS STANDARDS
For Hobbyists and Systems Developers

Sales to thousands of hobbyists over the past two years have proven the Tarbell Cassette Interface to be a microcomputer industry standard. Tarbell Electronics continues research and development to produce new and efficient components to fill hobbyists' changing needs.

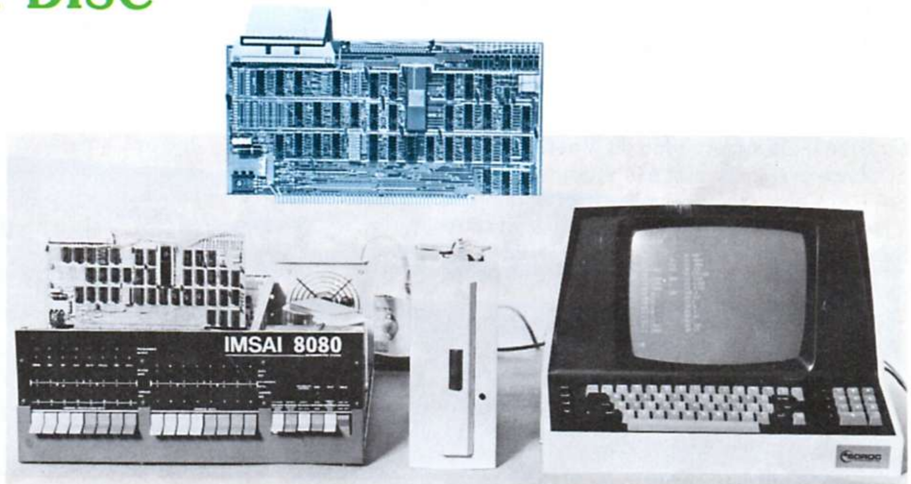


TARBELL CASSETTE INTERFACE

- Plugs directly into your IMSAI or ALTAIR*
- Fastest transfer rate: 187 (standard) to 540 bytes/second
- Extremely Reliable—Phase encoded (self-clocking)
- 4 Extra Status Lines, 4 Extra Control Lines
- 37-page manual included
- Device Code Selectable by DIP-switch
- Capable of Generating Kansas City tapes also
- No modification required on audio cassette recorder
- Complete kit $120, Assembled $175, Manual $4
- Full 6 month warranty on kit and assembled units

TARBELL FLOPPY DISC INTERFACE

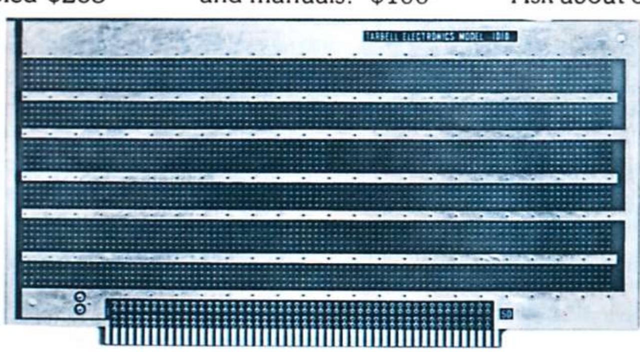
- Plugs directly into your IMSAI or ALTAIR* and handles up to 4 standard single drives in daisy-chain.
- Operates at standard 250K bits per second on normal disc format capacity of 256K bytes.
- Works with modified CP/M* Operating System and BASIC-E Compiler.
- Hardware includes 4 extra IC slots, built-in phantom bootstrap and on-board crystal clock. Uses WD 1771 LSI Chip.
- Full 6-month warranty and extensive documentation.
- PRICE:
  Kit $190 ...... Assembled $265



CP/M with BASIC-E
and manuals: $100

Compatible Disc Drives

Ask about our disc drives priced as low as $525.



TARBELL PROTOTYPE BOARD
Model 1010

- Gold plated edge pins
- Takes 33 14-pin ICs or
- Mix 40-pin, 18-pin, 16-pin and 14-pin ICs
- Location for 5 volt regulator
- Suitable for solder and wire wrap
- ALTAIR/IMSAI compatible

Price: $28.00

For fast, off the shelf delivery, all Tarbell Electronics products may be purchased from computer store dealers across the country. Or write Tarbell Electronics direct for complete information.

*ALTAIR is a trademark/tradename of MITS, Inc.
CP/M is a trademark/tradename of Digital Research

Tarbell Electronics
20620 South Leapwood Avenue, Suite P
Carson, California 90746
(213) 538-4251 T11

J. Tom Badgett
1917 Washington Street
Bluefield WV 24701

Trials and Tribulations

one businessman's micro blues

Smoky Mountain Aero. With about $200,000 per month in aircraft loans—not to mention gasoline sales, service, instruction, charter, rentals, tie-down and advertising—SMA uses a lot of business computer power.

"You see, I never had any intention of using my micro in my business; I just wanted it to play with," Jim Sexton tells me. I was surprised because other people had said Sexton's Maryville, Tennessee, flying service uses a sophisticated microcomputer business system.

"When I found out you could do some useful things with it," he continues, "I moved the unit from home out to the airport. I kept expanding it, and the next thing I knew I couldn't take it back home anymore."

Jim's background in electronics made him a natural microcomputer user. He was a radio-station engineer and owned an avionics service facility—among other accomplishments—before taking over Smoky Mountain Aero, one of the largest aircraft sales-and-service operations in eastern Tennessee. He still maintains his avionics shop and offers flight instruction and charter service besides.

Smoky Mountain Aero's primary micro system is built around an Imsai mainframe with Seals memory, a PROM board, Mits disk drive, a Lear Siegler terminal and an LA-36 DEC Writer. Smoky Mountain still shares a larger computer with a Maryville bank, but the micro handles mailing lists and letter writing, aircraft tie-down and maintenance records, aircraft insurance, advertising and sales duties. A second Mits system is in the works.

Choosing hardware wasn't easy; writing software is a continuing, frustrating job; keeping the machines on line is a big headache. Even so, the computer system is performing useful tasks for the business. "We've got enough stuff on it so we couldn't do without it now," Jim smiles wryly.

But, in spite of the smile, the Smoky Mountain Aero story is depressing at times, enough to make a potential microcomputer user fold up his memory cards and quit. On the other hand, the Smoky Mountain folk have a far-reaching view of the micro business. With this tone of ambivalence in mind let's start at the beginning.

Software Problems

Never one to do things in a small way, Jim Sexton visited various manufacturers and retailers around the country, flying to all the major cities in the East and Midwest for equipment demonstrations. The results were less than satisfying. He settled on an early Altair, then because of power supply limitations, bought an Imsai. Hungry for information to turn his new toy into a useful business tool, he continued visiting suppliers.

Due partly to a lack of software and apparent equipment problems, Jim began buying more hardware—a Processor Technology SOL, lots of memory, disks, etc.—but finally settled on the Imsai/Mits system he's currently using. The big problem was, and is, getting useful software to run on this system.

"I haven't met anyone in my travels—with the exception of Altair Software in Atlanta—who knows what they're doing," Jim recalls. Even Altair Software Distribution has its problems, though. In Jim's six visits to buy a word-processing package, the Altair Software people were unable to show him one that worked without file-link errors or disk-drive problems. "Every time they'd try to demonstrate it, the thing would switch off and come up with some kind of error. They've never been able to demonstrate a working package to me," he says.

Jim tried several books of programs, advertised to fill a plethora of business and hobby needs. He and his secretaries spent hours keying in the printed programs, only to discover that none worked properly. Based on a close analysis of two of the programs—Depreciation and Celestial Navigation—Jim believes the fault is with the printed program. "I can show you what is wrong with the programs," he says. "The formulas are wrong. I don't know where they got those formulas, but any basic finance book can show you the error."

Plus Ultra the Hobby Level?

Jim describes his experience with micros so far as "discouraging," but he hasn't given up on the idea of further utilizing his system to make his business more efficient. He believes, however, there will have to be some changes in hardware and software before personal computing will be more than a novelty appealing only to inveterate experimenters and hobbyists.

Jim has a strong electronics background, remember, but each time a glitch develops he has to go to Mits or Imsai for help. Engineers and programmers naturally want to know which memory locations he's using, how his software is configured, etc. . . . questions that only increase his frustration level.

"When I bought the computer I didn't take time to learn about memory locations and octal and hexadecimal and all this stuff; so they'd have to tell me on the telephone which switches to push, what to hit next. Then I'd read the panel and tell them which lights were lit, and they'd tell me what they thought was wrong."

Some recent hardware advances—cheaper memory, ROM programming, better disk systems—perhaps have eased some of the problems Jim suffered through in the beginning. Systems like the Commodore PET and the Radio Shack TRS-80 are welcome entries to the computer field, but Jim Sexton still believes the micro industry is too hardware oriented and suffers from a hobbyist mentality.

"The problem has been," he says, "that the micros have been designed for the experimenter—where cost is a major factor. I don't think it matters much what it costs, let's get one that'll do the job—make it reliable and easy to operate—and see if businesses can afford it."

The business market is where Sexton envisions the future of the micro industry. His business activities over the years have been varied: grocery stores, filling stations and other small operations that ran concurrently with other occupations. He'd buy a business that was in trouble, straighten it out, then sell it for a profit. This kind of transformation, he says, is relatively easy because the same problem usually exists: a lack of knowledge of what it is costing to do business. "Sometimes it is very hard to know what each sale actually costs."

Jim spends at least $500 per month on outside computer services from a local bank. He gets printouts like this every two weeks to help keep track of aircraft loans, business profits and tax information. The printouts also list aircraft operating records, maintenance information and pilot time. He'd like to use his own microcomputer for daily information, but he can't trust it.

Jim believes a computer system that worked, coupled with a reliable software package could encourage more people to get into small business and be successful. "I see the computer as a way of keeping them from failing," he says, but present systems would only "heighten the frustration level."

Where It Came From; Where It's Going

What initially sold Jim on microcomputers was a lot of potential at a reasonable price. He had checked out large computer systems from IBM, Burroughs and others, and the lowest-priced package he could get cost about $58,000. The basic hardware for that system was only $12K, but enough software and mass storage to do the job for Smoky Mountain Aero pushed the price up in a hurry.

"Over the years, the big companies, by withholding technology, have been able to sell a commodity at a very, very high price," Jim laments. "But what they're doing with their large systems isn't really out of reach of the micros." There's that ubiquitous software problem, though: so far, workable software that will run reliably on a micro has eluded Smoky Mountain Aero.

What is the seemingly elusive job Jim so desperately wants his micro system to do? It already is doing a great deal. In addition to typing original advertising letters and handling maintenance, tie-down and insurance records, the Smoky Mountain Aero computer keeps track of flight-training records for Veterans Administration-supported students. The VA requires a complicated series of records on each student to be filed each month. At Smoky Mountain there are at least 50 VA-certified students. "The computer reduces what used to take three days each month to a couple of hours," Jim says. "Mine has paid for itself in that alone if I could do nothing else with it." Still, Jim would like to see his system do more.

He'd like to eliminate use of the bank's computer entirely and switch to in-house processing for a sophisticated accounts-receivable and costing program for Smoky Mountain Aero.

"We have an accounts-receivable program written for use on our computer, but I'm afraid to start using it. If we get into any of these screw-ups—dropping bits and such—it could really ruin our accounting program, so we're still going with the bank because I trust the bank."

So, for now, instead of switching to an all-micro system, Smoky Mountain Aero is investing even more heavily in a time-share system. The micro will continue to handle the duties it already performs well: duties that wouldn't mess up the company's entire bookkeeping system should something go awry. They'll purchase several video terminals to access the bank's computer directly, perhaps using the micro system for some internal processing; that data could be fed to the large computer in a block.

"What I really want to do is get a daily profit/loss statement, which in this business is very, very difficult," Jim says. "In this industry, we are bothered by so many taxes and regulations that to employ someone at minimum wage to work the front counter and make decisions on what is taxable and what is not is difficult."

He'd like to use a terminal at the front desk tied to a computer programmed to make those decisions, keep track of state, federal and local taxes and print out profits at the end of each day. Currently, he gets this kind of information only every two weeks from the bank's computer. Smoky Mountain Aero also has a program to do *this* on the micro, "but," Jim reiterates, "I'm afraid to use it because of the unreliability of the computer."

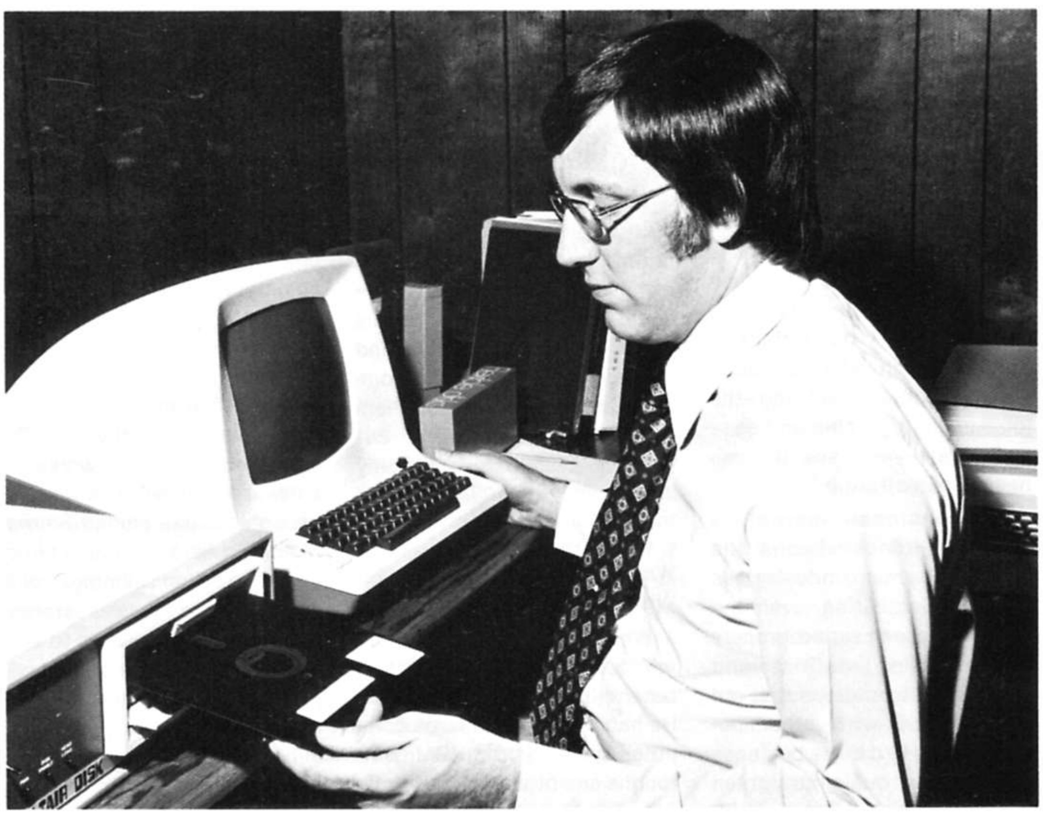
Jim Sexton has rejected the idea of buying or leasing a big machine. The cost is too high, and he already has committed around $8000 to micros. Besides, he *still* believes a microcomputer is capable of the job he wants done at a reasonable price. "I'm willing to pay a bunch of money to develop a computer system that'll work," he admits.

And he'd like to see better documentation with the equipment already available. One of the problems he had in the beginning was with his Altair 2SI/O boards. He couldn't make either of his two boards switch to the second port; so when he wanted to change from his ADM-3 video terminal to his DEC Writer, he'd turn off the computer, pull out the I/O board with the first port configured for the ADM-3 and plug in one set up for the printer. After a year of that—even with repeated calls to Mits for help—he wrote a program to overcome an apparent hardware problem.

"They kept telling me to read the instructions. Well, the instructions say if you're running the thing one way to put a certain switch up, but to change that for other conditions," Jim says. "They say, 'Where is the BASIC addressed?' and I have no idea. I say, 'How the hell do you tell?'"

With the kind of software Jim wants to run, an important capability is to read information off a disk or tape, update the information and put it back in the same place without destroying what is on either side. Documentation with his disk system is so unclear that so far he has been unable to make it work that way.

"They use terms I simply don't understand. If they'd charge me, say, $1800 for the disk system, then another thousand for a six-page booklet of operating instructions I could understand, then I'd be willing to pay the extra thousand bucks."



Jim Sexton has spent hundreds of hours trying to develop hardware and software for his business. His experience with micros, however, has been discouraging. "I'd be better off with a mini. That's what I'd recommend to anybody interested in their own business computer," he says. Even so, the investment has been a good one. The computer has paid for itself: "I'm not disappointed in my investment, it has been worth it. It just falls way short of what I'd like it to do."

Micros Should Get Down to Business

Well, I said in the beginning this story would be frustrating, discouraging, confusing. It is doubly so because I've talked with potential micro users and owners who say the same thing: they simply can't understand the instruction manuals. The manuals are either poorly written and illustrated, or written in such technical terms that only a designer or programmer with considerable experience could understand them.

"It seems the micro industry is just not conducive to business applications," Jim observes. "You really have to want to fool around with a computer, spend hours to make it work; I don't think businesses are willing to accept that.

A few businesses—Smoky Mountain Aero for one—have accepted it, but only because the owners have an interest in computers. Most businesses are interested only in what the computer can do for them, not how many bytes it stores, what chip it uses or how big the power supply is.

The American public is increasingly aware of computers' power, and businesses are expecting more and more from these machines they've heard so much about. The heavyweights in the computer industry aren't serving the small businessman. Micro hardware, on the other hand, is available to serve a wide mixture of business and personal needs at reasonable prices. Workable, versatile, affordable software to serve a variety of small operations hasn't yet arrived.

"The big-computer industry goes for a large General Motors-type company," says Jim. "They develop the software needs of that company, and if somebody else can fit it, fine. If they can't, it's just too bad. I'd hate to see the micro industry get to that point."

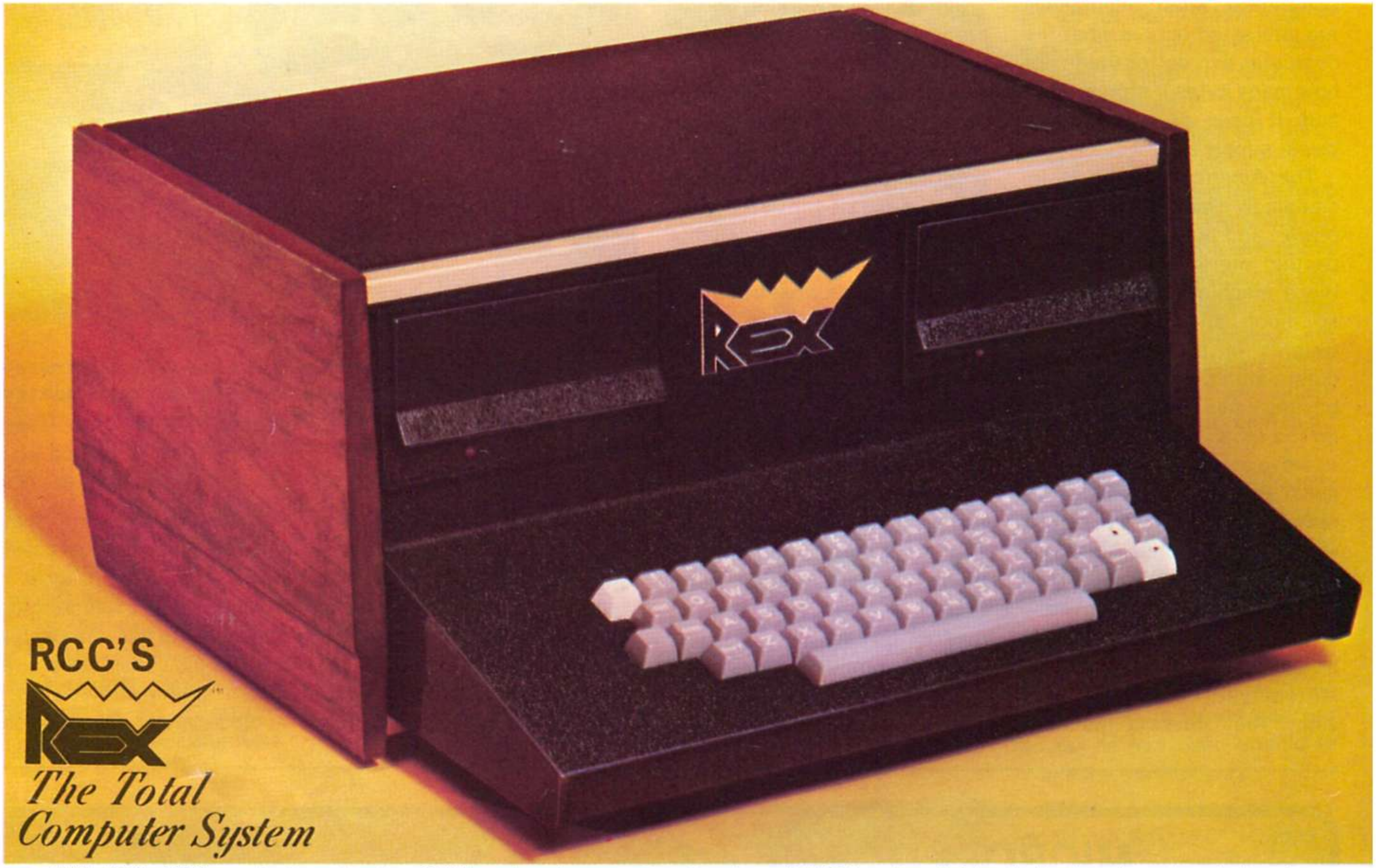
Jim Sexton's working on the problem. He has hired a full-time programmer in an effort to put his $8K worth of hardware to full use. Right now they're working on a parts-inventory program and trying to polish other aircraft-industry programs so they will run reliably on a variety of microcomputers. Already Jim has what he calls a "pretty good software collection" he hopes will benefit other Fixed Base Operators (FBO).

"There are 500 to 600 FBOs in this country, and maybe 50 of them would be interested in some of the things I have. That's not many units, but I'm trying to help aviation more than I am the computer industry," he says.

Small-computer stores are doing a booming business. They're selling the hardware faster than they can get it from manufacturers. But retailers and manufacturers might do well to note the experiences of businesses like Smoky Mountain Aero and take a dedicated interest in turning around what could be a disastrous trend. If software development doesn't keep up with technical development, we may be in for a user backlash that could set the industry on its ear. Jim Sexton is not the only businessman with mixed sentiments toward the microcomputer phenomenon. ■

Selling and servicing equipment such as this $475,000 Cessna 421 is only one facet of the Smoky Mountain Aero operation. Yet a single sale can involve complicated loan, trade-in and tax records, which can be made manageable only with a computer.

THE TOTAL COMPUTER SYSTEM



With RCC's REX you don't need to buy CPU boards and memory boards and parallel I/O boards and video display boards, and motherboards and serial I/O boards and disk interface boards and cabinets and chassis and keyboards and stuff!

REX is the complete small computer, designed for the user who wants to buy a system with enough power to run FORTRAN . . . or Disk BASIC . . . or Z80 Assembler .

The REX system costs **$2495**. Sure that's more than a kit, but look what you get!

Z80A CPU (4 MHz)	standard	
24K of RAM	standard/	expandable to 64K right on the motherboard
S-100 Bus	standard/	up to 5 slots available in basic cabinet. . .expansion cabinet available soon
Microfloppy drive	one standard, one optional/	record on both sides of the diskette; 75K bytes per surface
Bootstrap monitor	standard/	in a 2708 PROM. No switches; no buttons
Powerfail, 60 Hz timebase & vectored interrupts	standard	
Black & white video interface	standard/	128 character set, 24 line, 80 char/line, full software video control
DMA channels	standard/	control lines on the bus
Microfloppy interface	standard/	includes 8 bit parallel I/O port
ASR-33 type keyboard	standard/	optional typewriter keyboard with numeric 10 key pad
Walnut side-panel cabinet	standard	
15 Amp power supply	standard	

Your REX system is shipped assembled, tested and burned-in. You just add a video monitor (or an RF modulator and your home TV) and select one of our optional software packages ($195) and you are computing!

Accounting programs, analysis, games . . . no problem! Your basic REX system can handle them. Of course, we offer expansion capabilities too. Options like:

- ☐ Second microfloppy drive **$450**
- ☐ Programmable CRT character generator . . . . **$ 45**
- ☐ 12 inch CRT monitor (green phosphor, matching REX wood-side cabinet) **$212**
- ☐ 8K x 8 RAM expansion **$145**
- ☐ 16K x 8 RAM expansion **$285**
- ☐ Serial I/O channel (installed on initial order). . . . . . . **$ 35**
- ☐ RF modulator (for connecting REX to home TV) . . . **$ 35**
- ☐ Dot matrix printer (120 cps,80/96 column) **$850**
- ☐ Expansion cabinet for printer and two large floppy drives **$175**

The software is super! Take your choice for **$195**!

- ☐ ANSI FORTRAN IV, Disk Operating System, File Management System, Linking Loader, Text Editor, Utilities & Diagnostics
- ☐ Extended Disk BASIC, Disc Operating System, Text Editor & Diagnostics
- ☐ Z80 Macro Assembler, Disk Operating System, Text Editor & Diagnostics

See the REX system today at your local computer store. Or call RCC toll free at (800) 553-1863.

rcc™ **realistic controls corporation**

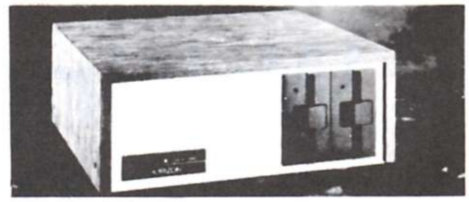
404 West 35th Street, Davenport, IA 52806
Toll Free (800) 553-1863 In Iowa (319) 386-4400

See the Rex at these dealers

ARIZONA		
Tempe	Computerworld, Inc.	(602) 944-6070
Tempe	Phoenix Group, Inc.	(602) 894-1129
CALIFORNIA		
Escondido	Business Enhancement Compuservice	(714) 741-6335
Huntington Beach	Cybernetics, Inc.	(714) 848-1922
Lawndale	Jade Computer Products	(213) 679-3313
San Diego	Byte Shop	(714) 565-8008
San Francisco	Computer Store of San Francisco	(415) 431-0604
San Jose	Byte Shop III	(408) 377-4685
San Rafael	Byte Shop–Computer Store	(415) 457-9311
Santa Clara	Byte Shop–Computer Store	(408) 249-4221
Sherman Oaks	Peoples Computer Shop	(213) 789-7514
Tarzana	Tech-Mart	(213) 344-0153
COLORADO		
Englewood	Byte Shop, Inc.	(303) 761-6232
IOWA		
Davenport	Computer Store of Davenport	(319) 386-3330
KANSAS/MISSOURI		
Overland Park	Personal Computer Center, Inc.	(913) 649-5942
MARYLAND		
Baltimore	Computer Workshop of Baltimore, Inc.	(301) 486-5350
Rockville	Computer Workshop, Inc.	(301) 468-0455
MICHIGAN		
Royal Oak	Computer Mart of Royal Oak	(313) 576-0900
MINNESOTA		
Minneapolis	Computer Depot	(612) 927-5601
MISSOURI		
Parkville	Computer Workshop of Kansas City, Inc.	(816) 741-5055
NEBRASKA		
Lincoln	Microtech, Inc.	(402) 423-8404
Omaha	Omaha Computer Store	(402) 592-3590
NEW YORK		
New York	Computer Mart of New York	(212) 686-7923
OHIO		
Lyons	Affordable Computers	(419) 923-2821
OREGON		
Beaverton	Byte Shop	(503) 644-2686
Portland	Byte Shop	(503) 223-3496
PENNSYLVANIA		
Pittsburgh	Computer Workshop of Pittsburgh, Inc.	(412) 327-0455
TEXAS		
Dallas	Computer Shops, Inc.	(214) 234-3412
Dallas	Rondure Company	(214) 630-4621
Houston	Interactive Computers	(713) 772-5257
San Antonio	The Computer Shop	(512) 828-0553
San Antonio	Computer Solutions	(512) 828-1455
VIRGINIA		
Springfield	Computer Workshop of Northern Virginia, Inc.	(703) 321-9047
WASHINGTON		
Bellevue	Byte Shop	(206) 746-0651
Bellevue	The Microcomputer Center	(206) 455-3710
WISCONSIN		
Janesville	Austin Computers	(608) 752-6002

rcc™ *realistic controls corporation*

Look To The Horizon.™ The North Star HORIZON Computer.



HORIZON™

To begin programming in extended BASIC, merely add a CRT or hard-copy terminal. HORIZON-1 includes a Z80A processor, 16K RAM, minifloppy TM disk and 12-slot S-100 motherboard with serial terminal interface—all standard equipment.

And, Look To Computer Enterprises For The Lowest Prices & Fastest Delivery On North Star Horizon Computer System:

	Credit Card Price	Cash Discount Price
Horizon 1 System (one drive) kit	$1497	$1439
Horizon 1 System (one drive) assembled	$1777	$1709
Horizon 2 System (two drives) kit	$1871	$1799
Horizon 2 System (two drives) assembled	$2199	$2114

ALSO: Run TDL Software On Your Horizon...

TDL Package A with Super Basic	$ 228	$ 219

For Your Serial Terminal:

Lear Siegler ADM-3A (kit)	$ 728	$ 700
Lear Siegler ADM-3A (assembled)	$ 858	$ 825

Call Or Write Today For Computer Enterprises' FREE Spring Computerlogue!

IMSAI PCS-80/30 kit	$1097	$1055
IMSAI PCS-80/15 kit	$ 748	$ 719
IMSAI VIO-C kit	$ 297	$ 286
TDL ZPU Board (assembled)	$ 183	$ 176
TDL XITAN ALPHA 1.5 kit	$ 812	$ 781
Seals 8k 250ns RAM (assembled)	$ 203	$ 195
Dynabyte 16k 250ns Static RAM (assembled)	$ 577	$ 555
Cromemco Z2D kit	$1399	$1345
Cromemco Bytesaver less Proms kit	$ 136	$ 131

Shipping charges: $10 per CPU on larger units; $1.50 per kit. $2.00 min. per order.
Delivery is stock to 30 days on most items. Shipment is immediate for payment by cashier's check, money order or charge card. Allow 3 weeks for personal checks to clear. N.Y. State residents add approp. sales tax. Availability, prices and specs may change without notice.

computer enterprises™

Operating Hours:
M-W 10-5 E.S.T.
TH-F 10-9 E.S.T.
Closed Sat. & Sun.

P. O. Box 71 Fayetteville, N.Y. 13066
Phone (315) 637-6208 Today!

Robert Baker
15 Windsor Drive
Atco NJ 08004

Writing Diagnostic Routines

while your machine is running

Ever since personal computer systems first caught on, the area of diagnostic software has been overlooked. Few companies producing kits and hobby systems offer diagnostic programs to test and debug their products should a problem be suspected. Possibly such programs haven't been offered yet because of the degree of customizing each user performs while assembling his or her system. Because of competition in the market, most systems are a conglomeration of bits and pieces interconnected for a particular application. Thus, most hobbyists are forced to write their own diagnostic or maintenance programs (usually after a major problem develops) without really knowing what they're doing. It's very difficult to debug software if the hardware is not working properly!

On mini and larger size computers, the manufacturer usually provides various diagnostic programs to be run at regular intervals by the user as a form of preventive maintenance. The programs are written to detect minor faults before they degrade system operation, and to help isolate and debug major problems when they occur.

Diagnostic Methods

One of two approaches is generally taken for writing and using diagnostic programs. A *bottom up* approach starts by testing the smallest entity in the system then using that proven-good device to test the next device in the system until all devices have been tested. Individual components are then tested in clusters or subsystems, and finally the entire system is tested, or exercised, as a whole.

A *top down* approach, on the other hand, starts by running a system exerciser to test all devices at once and isolate a problem to a given subsystem. More detailed, device-dependent programs are then run for the particular faulty device or subsystem to further isolate and help debug the problem. Once the problem has been corrected and the device-dependent tests are passed, the system exerciser can be run again to verify that that was the only system fault.

A bottom up approach requires the least amount of working hardware to be useful, but a top down approach takes less time to isolate a given system fault; so there are trade-offs. For either approach, the actual programs could be similar, depending on the system and the application.

Writing A Test Program

Why not write a collection of test programs while you have a working system to try them on? Debug your programs thoroughly when writing them so you're sure that any problems detected are caused by hardware and not software. Try a bottom up approach first as this should make the programs easier to write. Start out with a few simple programs to check the CPU machine instructions, checking operands, condition codes, etc. Then check data paths to and from the CPU and the various control logic, trying different bit patterns to check for shorted lines. For convenience, you may want to create some of these programs in ROM and have them permanently available. Loading programs would require a major portion of the CPU to be in working order, so using ROM would eliminate that problem.

A quick memory check can verify that RAM memory is working correctly by writing all zeros and all ones to each location, reading it back, and comparing the data. You may want to check another memory pattern such as alternating ones and zeros (10101010) as well as checking memory addressing logic by insuring a test pattern had not been written into another location.

After the CPU and memory have been tested, you can then proceed to test any other devices you may have in your particular system. Test each device separately and thoroughly before going to the next. For starters, try a program to test your CRT display or video terminal with:

- a character generator check, full lines of each character.
- display memory test (swirl pattern). The first line is a full character set. Each line after the first starts with the next letter in the character set after that used on the line above it. Therefore, each character will be on a diagonal, and will appear in each storage location as the test is run, and the display scrolls. For example:

```
abcdefg . . . . . . . . . . . . .
bcdefg  . . . . . . . . . . . . .
cdefg  . . . . . . . . . . . . . .
defg  . . . . . . . . . . . . . . .
```

Other patterns can be added to test special features, etc., depending on the particular display. To test a keyboard you can try a program that:

- displays on your terminal or CRT the code for the key depressed and the actual character.
- asks you to type each character in a set sequence and checks the code received.

Another useful test could print continuous lines of any character typed in by the user. When another is typed in, the printer would change to that character. Similar tests can be written to check the particular features or functions of other displays and terminals. Use easy-to-recognize patterns on printers or terminals and keep the tests simple!

If you have tape drives, cassettes, floppy drives, optical readers or joysticks, don't forget to test them also. Test every device in your system thoroughly, one at a time. Later you can add a simple exerciser to get everything working at once and check for device interactions. My advice is to save this for later when you start to get a feel for what you really want to accomplish and how you want to control your test programs.

Make It Useful

For whatever devices are being tested, certain basic features should be included for your own convenience and increased usefulness of the programs. Each test within a program should give a clearly defined indication when an error is detected. This can be an error message on a terminal or simply a machine stop at a specific address. If error stops are used, separate halts or stops should be used for each possible error so the address at which the machine stops will indicate *what* error was detected. For added convenience you may even want to generate an error dictionary to list and describe each possible error halt and give some possible causes or cures. Also, keep a log of the errors and causes detected by your programs for later reference. They may save you from debugging the same problem again several months or years later.

Another desirable feature should be the capability to loop on any test for scoping purposes, possibly with a sync pulse generated at the start of each pass. Other features can be added as desired, depending on your particular system and how you want to test it. Try to keep it simple but flexible, and easy to use.

Plan ahead and prepare yourself for the inevitable. Sooner or later you're bound to have a hardware problem, and your local TV repairman will probably not be able to help you, let alone know what you're talking about. An even better idea: Run the tests periodically and catch small problems before they become major ones. Your time spent writing diagnostic programs for your system now will be repaid many times in the future. ■

Experiments in Software

serial to parallel conversion

Dan Stogdill
182 Victoria Street
St. Marys Ontario
Canada N0M 2V0

As most hobbyists are aware, microcomputers generally employ parallel data to carry out their internal transactions. That is, the CPU expects all bits of a given data word to exist simultaneously for a finite length of time, one bit on each of the parallel wires of the data bus. Nevertheless, in order to communicate with the outside world, this parallel data must frequently be converted to serial form in which each bit of a given data word is transferred sequentially between the microcomputer and its external devices. Technology has provided us with at least two hardware devices (the UART and the ACIA) designed to achieve this end. Instead of hardware, however, software may also be employed to attain the same results.

Although there have been several articles on this subject in the literature, most of them contain a cautionary note related to the parameters associated with the software timing loops. By way of illustration, the typical routine which converts incoming serial data to parallel form does so by testing the middle of each bit in order to determine whether it represents a binary 1 or 0. Following examination of the center of one bit, the program will loop through a time delay routine for a fixed length of time. This length of time corresponds to the duration of one bit. Upon exiting from the loop, the program tests the next bit, loops . . . and so on. The requisite loops are dependent on the speed with which the CPU executes its instructions (i.e., does it have a 1 MHz clock, 2 MHz clock, etc?) and by the access time of the memory. With slower memories it is general practice to insert one or more wait states before the memory is read. As a consequence, to some extent serial-to-parallel conversion routines tend to be machine or system specific.

In this article, I intend to describe a simple serial-to-parallel routine, READ, and a

Symbolic Address	Location	Machine Code	Mnemonic	Comments
CALIBRATE	000-200	041	LXI H	Initialize the storage
	201	100		location for the
	202	000		critical value
TST	203	333	IN	Input the parallel
	204	005		port
	205	346	ANI	Set up a
	206	001		mask
	207	302	JNZ	If no start bit
	210	203	TST	go back to TST
	211	000		
	212	001	LXI B	Initialize the B-C
	213	000		register pair to
	214	000		zero
	215	315	CALL	Call subroutine
	216	261		TIMER 1
	217	000		
	220	161	MOV M,C	Store C register
	221	043	INX H	Bump the pointer
	222	160	MOV M,B	Store B register
	223	166	HLT	Halt

	---			Unused memory space

TIMER 1	000-261	003	INX B	Increment B-C pair
	262	333	IN	Input the parallel
	263	005		port
	264	346	ANI	Set up a
	265	001		mask
	266	177	MOV A,A	Extra-parallels TIMER 2
	267	267	ORA A	Extra-parallels TIMER 2
	270	312	JZ	If no data bit
	271	261		go back to TIMER 1
	272	000		
	273	311	RET	Return to CALIBRATE

Table 1. This routine determines the critical value and stores it in addresses 100 and 101 (octal) on page 0.

parallel-to-serial routine, WRITE, which allows my Altair 8800 (through one of the *parallel* ports on a Processor Technology 3P+S interface) to converse with my CRT terminal which employs an RS232C *serial* I/O. Of greater importance, I will outline a simple technique by which anyone can determine the values of the loop counters for any particular system.

Some Preliminaries

Fig. 1 illustrates the simple connections made to the 3P+S in order to boost the TTL signals at the parallel I/O port to RS232C level capable of interfacing with my terminal. As can be seen, for outputting data, one output line of a parallel I/O port (available at J1 pin A) is fed back to the input of an unused gate of IC 5, the 1488 which normally boosts the 3P+S's UART output to RS232C level. For inputting data, one input line of a parallel port (available at J2 pin Z) is connected to the

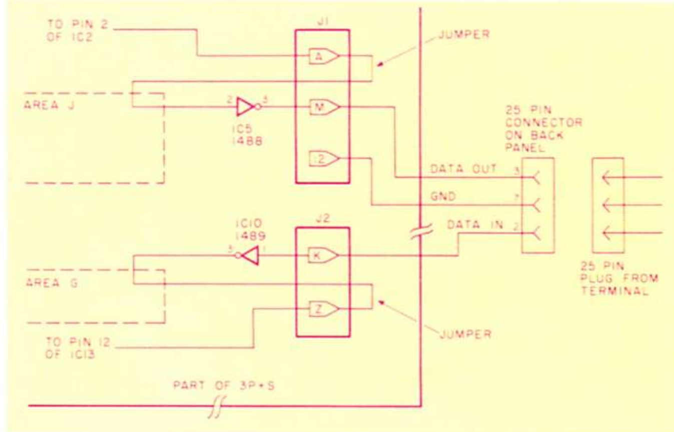


Fig. 1. The simple modifications to the Processor Technology 3P+S interface.

Symbolic Address	Location	Machine Code	Mnemonic	Comments
READ	000-000	305	PUSH B	Save these
	001	325	PUSH D	registers
	002	345	PUSH H	
	003	021	LXI D	Set up number
	004	010		of word bits in reg. E
	005	000		Clear reg. D
TESTR	006	333	IN	Input the
	007	005		parallel port
	010	346	ANI	Set up a
	011	001		mask
	012	302	JNZ	If no start bit
	013	006		go back to TESTR
	014	000		
	015	001	LXI B	Initialize B-C pair
VALU 1	016	224		with loop counter
	017	002		value
	020	315	CALL	Call TIMER 2 and
	021	061		loop for awhile
	022	000		
NEXT	023	333	IN	Input the parallel
	024	005		and fetch a data bit
	025	346	ANI	Set up
	026	001		mask
	027	202	ADD D	Add D reg. to A reg.
	030	017	RRC	Shift reg. A to right
	031	127	MOV D,A	Save byte in reg. D
	032	001	LXI B	Initialize B-C pair
VALU 2	033	270		with loop counter
	034	001		value
	035	315	CALL	Call TIMER 2
	036	061		and loop
	037	000		for awhile
	040	035	DCR E	Decrement bit counter
	041	302	JNZ	Fetched all bits?
	042	023		No. Go back to next and get
	043	000		another bit
	044	172	MOV A,D	Put word in Acc.
	045	247	ANA A	Set the flags
	046	342	JPO	If parity is odd
	047	057		jump to WRONG
	050	000		
	051	346	ANI	Strip the parity
	052	177		bit
	053	341	POP H	Retrieve the
	054	321	POP D	previously stored
	055	301	POP B	stored registers
	056	311	RET	Return to calling program
WRONG	057	166	HLT	Safety halt
	060	000	NOP	Unused
TIMER 2	000-061	013	DCX B	Decrement loop counter
	062	333	IN	Extra-parallels TIMER 1
	063	005		Extra-parallels TIMER 1
	064	346	ANI	Extra-parallels TIMER 1
	065	001		Extra-parallels TIMER 1
	066	171	MOV A,C	Fetch counter value
	067	260	ORA B	Are B and C zero?
	070	302	JNZ	No. keep looping
	071	061		
	072	000		
	073	311	RET	Return to calling program

Table 2. This program converts incoming serial data to parallel format and checks for even parity. (Note that on my system I obtained a critical value of 270-001, as shown by VALU 2.)

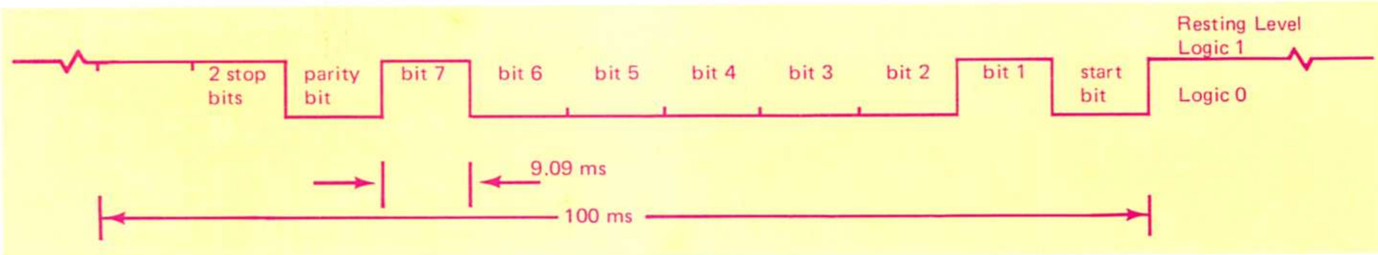


Fig. 2. The bit composition of the ASCII encoded character A (capital), and its associated timing values when transmitted at a rate of 110 baud. Note that the resting level is always logic 1, while the start bit is always logic 0.

output of an unused gate of IC 10, the 1489 which converts the incoming RS232C signal to TTL level compatible with the UART's input.

Owners of other parallel I/O boards and/or current loop type terminals can also interface them readily. Back issues of *73* and *Kilobaud* contain a number of circuits based on the hardware UART. Simply borrow that part of the circuit which is responsible for changing the UART's I/O signals to the appropriate drive level consistent with your particular peripheral.

The Critical Value

Fig. 2 depicts the bit composition of the ASCII character A (capital), and its timing values when it is transmitted serially at a rate of 110 baud. Note that each bit occupies a time of 9.09 ms and that the resting level is always logic 1, while the start bit is logic 0. In addition, observe that the first data bit of the A is always a logic 1 . . . this is very important in determining what I call the critical value.

Table 1 contains a short program labeled CALIBRATE. This program operates by testing for the start of transmission of the character A (for example) when it is transmitted by the terminal. Upon detecting the logic 0 start bit (see Fig. 1), it calls up a subroutine TIMER 1, which alternately incre-

Symbolic Address	Location	Machine Code	Mnemonic	Comments
WRITE	000-100	305	PUSH B	Save these
	101	325	PUSH D	registers
	102	345	PUSH H	
	103	247	ANA A	Set the flags
	104	352	JPE	If even parity
	105	111		in Acc. then
	106	000		skip to EVEN
	107	356	XRI	Make Acc.
	110	200		even parity
EVEN	111	127	MOV D,A	Save the word in reg. D
	112	036	MVI E	Set up number of
	113	010		word bits in reg. E
	114	257	XRA A	Zero the Acc.
	115	323	OUT	Out this as a start
	116	005		bit to the terminal
	117	001	LXI B	Initialize the
VALU 2	120	270		loop counter
	121	001		
	122	315	CALL	Call TIMER 2
	123	061		and loop
	124	000		
OUTIT	125	172	MOV A,D	Move the word to Acc.
	126	323	OUT	Output a character
	127	005		to the terminal
	130	017	RRC	Shift reg. A to right
	131	127	MOV D,A	Save the word in reg. D
	132	001	LXI B	Initialize the
	133	270		loop counter
	134	001		
	135	315	CALL	Call TIMER 2
	136	061		and loop
	137	000		
	140	035	DCR E	Decrement bit counter
	141	302	JNZ	Outed all bits?
	142	125		No. Go back to outit and
	143	000		keep outputting bits
	144	076	MVI A	Put a 1 in Acc.
	145	001		Use as a stop bit
	146	323	OUT	Output the stop bits
	147	005		to the terminal
	150	001	LXI B	Initialize the
VALU 3	151	160		loop counter
	152	003		for two stop bit lengths
	153	315	CALL	Call TIMER 2
	154	061		and loop
	155	000		
	156	341	POP H	Retrieve the previously
	157	321	POP D	pushed registers
	160	301	POP B	
	161	170	MOV A,B	Restore the Acc.
	162	311	RET	Return to the main program

Table 3. This program converts parallel data to serial format and outputs it to the terminal.

ments the value in the B-C register pair and tests for the beginning of the first data bit, a logic 1 (see Fig. 1). Upon detecting the logic 1 start bit, TIMER 1 is exited with a relative value of the bit length in the B-C register pair. Should you run the program several times, you will see that the critical value stored in addresses 100 and 101 (octal) will remain quite stable. (Note that address 100 contains the least significant bits and that address 101 contains the most significant.) If you have access to a terminal with adjustable baud rates, you will notice that the critical value becomes progressively smaller as the baud rate increases.

The READ Program

Table 2 contains the program READ which converts incoming serial data to parallel format. It assumes that the parallel I/O is at address 005 and, furthermore, that the serial data is entering through line zero of the input port (DI0). VALU 2 is the previously obtained critical value. VALU 1 represents the critical value converted to decimal, multiplied by 1.5, and converted back to octal.

TIMER 2 may appear to contain some irrelevant instructions. Not really. TIMER 1 and TIMER 2 were fashioned in such a manner that the critical value obtained with TIMER 1 would be directly applicable with TIMER 2; their respective instructions were juggled around while insuring timing compatibility between the two routines.

The READ program tests for the beginning of the start bit and then circulates through the loop for a time equal to 1.5 bit times (governed by VALU 1). At this point, the middle of the first data bit is tested. Subsequently, TIMER 2 loops for 1 bit length (governed by VALU 2, the critical value), exiting near the center (hopefully) of each successive bit until all bits have been tested and the word assembled in parallel.

The WRITE Program

Table 3 contains the program WRITE, which converts a parallel data word to serial format and outputs it to the terminal. It also assumes that the I/O port is located at address 005 and that serial transmission to the terminal is occurring through line zero of the parallel outport (DO0). VALU 3 produces the stop bits and is equal to the critical value converted to decimal, multiplied by 2, and converted back to octal.

Conclusions

Both routines assume that they will be called up by other programs to provide I/O services; as such, the states of all registers except A and PSW are saved on the stack so as to not interfere with other operations. As provided, both routines return to the main calling program with the data word in the accumulator. In addition, WRITE assumes that the data word is in registers A and B on being called. Lastly the routines provide for parity generation and checking (even parity, in the present case).

While described from a viewpoint of being used with a terminal, the routines should lend themselves for use as cassette I/O routines in concert with appropriate encoding/decoding schemes.

The software connoisseurs among us could make them more efficient memory-wise. In addition, I am sure that CALIBRATE and READ/WRITE could be combined in such a way as to make the whole process automatic in nature.

What initially started out as an experimental project is now in everyday use on my system. By all means experiment with them and have fun. As ever, if you have any questions drop me a line and I will try to help out. Keep on computing. ■

Peter Stark
Box 209
Mt. Kisco NY 10549

Computer Math Primer

beginner's introduction to number systems

Once you understand a little about hex, that will make computers a little less magical and bring them down to earth. So here goes.

Hex is short for *hexadecimal*, one of four number systems used with computers. The other three are binary, octal and decimal. Each is based on a particular number: binary uses the *base* 2, octal the base 8, decimal the base 10 and hexadecimal the base 16. Table 1 shows their relationship.

Notice that binary numbers only use 0 or 1; there are no digits larger than 1. So, the next number after 0 and 1 is 10 (0010), since we have to skip over 2, 3 and so on. (For most purposes, 10 and 0010 are the same; the beginning zeros don't change things.) After 10, binary numbers skip to 11, then skip to 100 and 101, etc. Since 101 is the fifth number after 0, it stands for 5.

There are no 8s or 9s in octal numbers, so after 7 comes 10; after 17, 20; after 77, 100.

Decimal numbers go from 0 through 9. That's only ten different digits, so after 9 we have to start doubling up; after 9 comes 10, and so on.

The hexadecimal number system has 16 different digits. The first ten are the same as the ten digits of the decimal system — plus there are six more.

I think whoever devised hex made a big mistake here. Having six new digits, he should have invented six new symbols for them. Instead, he simply gave his six new digits old symbols: A, B, C, D, E and F. Consequently, we have a number system that goes from 0 to F. (After F, by the way, you have to start doubling up as in decimal. After F comes 10, which corresponds to 16 in decimal.)

Computers use binary numbers for their *internal* operations. Octal and hex numbers are used by the people who use computers (not computers themselves) because they are easier to see and interpret (A3, for example, is easier to read than 10100011). Computer people use octal or hex numbers rather than decimal because their conversion to and from binary is easy and fast. (Use of these number systems constitutes *external* operations.)

Whether octal or hex is used depends partially on the length of the binary numbers used, and partially on personal preference. Computers handle binary numbers of a fixed length, called a *word length*. Most hobby computers use a word length of 8, meaning they handle binary numbers in groups of eight digits. If the binary word length is divisible by three, octal is generally used; if it is divisible by four, hex is employed. If it is divisible by both three and four, or by neither, then it's a matter of choice.

So, most hobby computers with a word length of 8 use hex (externally), although there are exceptions: the 8008 CPU uses octal, as does Heathkit's new H8. The 12-binary-digit (12-bit — a bit is a binary digit) Intersil uses octal.

Most hex or octal numbers used with small computers are small two-digit hex or three-digit octal numbers. Let's use those as examples of how to do magic with them. (Assume only positive integers for a starter.)

Converting Hex to Binary

For each hex digit, replace it by its *four-bit* binary equivalent from the table. For example, to convert hex A3, replace A by 1010 and 3 by 0011 for a binary result of 10100011.

Octal to Binary

For each octal digit, replace it by the rightmost three bits of the corresponding binary number in the table. To convert octal 243, for example, replace the 2 by 010, the 4 by 100 and the 3 by 011. The complete binary number is 010100011. Now, for a second trick: This procedure gives you a total of nine bits, whereas most small computers need only eight. Fortunately, the leftmost bit will usually be a zero, and so

Binary	Octal	Decimal	Hex
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F

Table 1.

can be crossed out, giving the final eight-bit answer of 10100011.

Binary to Hex

Starting from the right, separate the binary digits into groups of four; replace each group by its hex equivalent. Binary 01100100 would be split into 0110 and 0100. 0110 is replaced by 6 and 0100 by 4, giving hex 64.

This conversion is easy if the number of bits is any multiple of 4. If not, you must add zeros at the left until it is. For instance, to convert 11011 you first add three zeros to make it 00011011, then split it up into 0001 and 1011, and finally convert to 1B.

Binary to Octal

This is the same as the hex conversion except that we use groups of three bits. For example, 01100100 is an eight-bit number, and eight bits cannot be separated into groups of three; so we add an extra zero to make it 001100100. We can then break it up into 001-100-100. We can convert each group into octal using the table if we note that 001 is the same as 0001, or an octal 1; 100 is the same as 0100 or an octal 4. Thus, binary 001100100 is octal 144.

(Just a reminder: We are working only with positive whole numbers (integers) now. Negative numbers or fractions are a different ball game.)

Binary to Decimal

The easiest way to do this is to convert the binary number to octal or hex first and go to decimal from there. If you insist on doing it directly, then here's how.

In a binary number, each bit has a specific value; to convert, you have to multiply each bit by its value and add the results. Starting from the *right*, the values are 1, 2, 4, 8, 16 and so on – each value is twice the preceding value.

To convert the binary 1011, transcribe the number (spread it out), and under each bit write its value.

1	0	1	1
8	4	2	1

Remember to start with a value of 1 at the *right*. Now multiply each bit by its value, like this:

1	0	1	1
x8	x4	x2	x1
8	0	2	1

Finally, add 8+0+2+1=11. (Check the table and you'll see that 1011 in binary is 11 in decimal.)

To see how this works for larger numbers, see Example 1. (With a little practice, you can skip multiplying by zero.) Add the results (128+32+16+4) and you'll see that binary 10110100 converts to 180 in decimal.

Octal to Decimal

This conversion is the same as binary-to-decimal except that the values of each digit are 1, 8, 64, 512 and so on – each value is eight times more than the one before it. To convert octal 264, for example, use the preceding technique with the new values.

2	6	4
x64	x8	x1
128	48	4

Now add up 128+48+4=180.

Hex to Decimal

Same as before, except the digit values are now 1, 16, 256, 4096, etc. (remember to start with 1 at the right) – each value increases 16 times. Hex 2C4 converts like this:

2	C(12)	4
x256	x 16	x1
512	192	4

512+192+4=708.

Decimal to Binary

Convert to octal or hex first; then convert the result to binary. There is a direct way, but it's likely to take longer, and you will probably make a mistake – so don't bother.

Decimal to Octal

This conversion is done by dividing the decimal number by 8 and saving the remainders. Keep doing this until you get 0, and then put the remainders together backwards.

This sounds crazy until you see how it's done. Let's say you want to convert 180 from decimal to octal. Start by dividing 180 by 8. Don't use your calculator for this because you will just get an answer of 22.5, and then you won't know what to do. Use pencil and paper.

Eight goes into 180 twenty-two times, with a remainder of 4. (After a while you'll figure out how to use your calculator for this, too.) Put away the 4, and divide the quotient by 8.

Eight goes into 22 twice, with a remainder of 6. Save the 6, and divide 2 by 8.

Eight goes into 2 zero times, with a remainder of 2. Since we are down to a quotient of 0, we can stop dividing by 8.

Now take the three remainders (4, 6 and 2) and write them backwards: 264. This is your octal number.

Decimal to Hex

This conversion is the same as decimal-to-octal, except you divide by 16. For example, to convert 180 to hex, start by dividing 180 by 16.

Sixteen goes into 180 eleven times, with a remainder of 4. Save the 4 and repeat.

Sixteen goes into 11 zero times, with a remainder of 11. Again, we stop dividing when we get a quotient of 0.

Write the remainders down backwards, but convert any remainder above 9 to its hex digit. In this case, the 11 converts to B, and the hex answer is B4.

Complements

A complement is an opposite; in the case of computers, complements are used for negative numbers. There are two kinds of complements: *one's complements* and *two's complements*. The one's complement is easy to find, but the two's complement is generally used.

1	0	1	1	0	1	0	0
x128	x64	x32	x16	x8	x4	x2	x1
128		32	16		4		

Example 1.

Converting a binary number to one's complement. Write down the binary number. Then *invert* each bit – that is, change each 1 to a 0 and each 0 to a 1. For example, the one's complement of 10110 is 01001.

Converting an octal number to one's complement. Write the octal number. Then *above* each digit put a 7. Now subtract each bottom digit from the top digit.

To convert the octal 0145, for example, you proceed like this:

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

Remember that in the binary number that's actually in your computer, each 0 is being inverted into a 1 as you complement. Any *extra* zeros you put in will produce extra ones in the complement. For example, octal 5 is binary 101. But it is also 0101, 00101, 000101, etc., since putting extra zeros in front of a binary number does not change it. But look what happens if you try to get the one's complement (Fig. 1).

An octal 5 can have many different complements; but notice that the only difference between them is the presence of extra ones at the left. The solution is to use only as many ones at the left as will fit the word length of the computer being used. For example, in an eight-bit computer the complement of 5 would be 11111010 binary, or 372 octal.

So, whenever you find the complement of *any* number, always be sure to keep in mind the word length of your computer, and modify the answer to fit your word length. In the case of hobby computers, this problem usually arises on either the Heath H8 computer or any 8008 system, which use octal with an eight-bit word length. Since the leftmost octal digit of any octal number on these computers only stands for two binary digits, the largest it can be is octal 3 (or binary 11). Hence, any complement that starts with a digit *greater* than 3 is wrong. The usual trick is to subtract a 4 from the leftmost digit.

7	7	7	7	7	8	0	0
-0	-2	-3	-0	-7	-5	-0	-0
7	5	4	7	0	3	0	0

Example 2.

Suppose you want the one's complement of 005. If you follow the rule for converting, you get

777
-005
772

Since the leftmost digit is greater than 3, there is an extra bit. Remove it by subtracting 4 from it, so the actual complement is 372.

Converting a hex number to one's complement. The rule is the same as for octal numbers, except that we write a 15 above each digit and convert hex digits to and from decimal.

The one's complement of hex 68 is hex 97.

15	15
- 6	- 8
9	7

The one's complement of hex 9E is hex 61; we have to convert E to 14:

15	15
- 9	- E (14)
6	1

The one's complement of hex 61 is hex 9E; this time we have to convert 14 to a hex E:

15	15
- 6	- 1
9	14 (E)

The same warnings about extra ones in the complement apply here as when using octal numbers; but we don't usually have to worry about it because in most computer systems the number of bits matches the hex digits exactly. For example, the two hex digits used in eight-bit computers like the 8080 or 6800 match the word length exactly.

Converting numbers in one's complement to two's complement. As mentioned before, most systems use two's rather than one's complements. It's easy to convert from one's to two's complement: add 1. If the one's complement of some number is 110, the two's complement is 111; if it's 61, the two's complement is 62; if it's 9B, the two's complement is 9C – adding 1 to B (which is 11) makes it C (12).

Be careful how you add 1 – it has to be done right. For example, if the one's complement is a binary 101, adding 1 does not give you 102 because a 2 is not allowed in binary! 101 plus 1 is 110 (refer to the table).

Although this is irrelevant anyway since there are other ways of converting, it is of some interest since many microprocessors convert to the two's complement by first finding the one's complement and then adding a 1. For instance, the Intersil 6100 has a CIA (complement and increment accumulator) instruction. (*Increment means to add one.*)

Converting a binary number to its two's complement. Write the binary number. Now find the rightmost 1 and put a vertical line just to the left of it. Invert all bits to the left of this line. Leave the bits to the right of the line unchanged.

Convert the binary number 10110 thus:

1 0 1	1 0
0 1 0	1 0
invert	leave alone

The two's complement of the eight-bit number 00000101 is 11111011:

0 0 0 0 0 1 0	1
1 1 1 1 1 0 1	1
invert	

Converting an octal number to its two's complement. Write the octal number and see whether it has any zeros at its *right* end (ignore zeros in the middle or at the left). If so, put a zero *above* each zero at the right. For instance, if you wanted to convert the octal number 02307500, you would now have

						0	0
0	2	3	0	7	5	0	0

Continue from the right and put an 8 above the next digit and a 7 above each of the others. Finally, subtract each digit from the one above it (see Example 2). The two's complement in this case is 75470300.

Just one warning: Everything we said about extra ones in the one's complement conversion applies here, too. For instance, in an eight-bit computer the complement of 005 would be 373, not 773.

If you find this method too hard to remember, you can always convert your octal number to binary, find the two's complement of that, and then convert that back to octal.

Converting a hex number to its two's complement. Look at the hex number to see whether it has any zeros on the right end (ignore zeros in the middle or at the left). If it does, put a zero above

Binary Number	Binary Complement	Octal Complement
101	010	2
0101	1010	12
00101	11010	32
000101	111010	72

Fig. 1.

each of these rightmost zeros. To convert C0B0, you would write:

			0
C	0	B	0

Continue from the right and write the number 16 above the rightmost nonzero digit of the hex number; write 15 above each of the other digits. Finally, subtract each of the hex digits from the number above it, converting from letters to numbers — or back if needed. C0B0 converts to 3F50 (Example 3).

15	15	16	0
- C (12)	- 0	- B (11)	-9
3	15 (F)	5	0

Example 3.

As another example, the two's complement of hex 05 is FB:

15	16
- 0	- 5
15 (F)	11 (B)

By the way, the two's complement of a two's complement is the original number; the two's complement of FB is 05:

15	16
-F (15)	-B (11)
0	5

Converting Decimal to BCD

Many computers allow calculations to be done in binary coded decimal (BCD) rather than only in binary. (BCD is a combination of binary and decimal.) Converting decimal to BCD is performed in the same way as converting hex to binary: Replace each decimal digit by its *four-bit* binary equivalent from the table. To convert decimal 93, replace 9 by 1001 and 3 by 0011 to get 10010011.

Notice that this result is different from the 01011101 you would get if you converted 93 to binary. In converting to binary, you convert an entire decimal number at once; in converting to BCD, you convert only one digit at a time.

Watch out for one big area of confusion. If you convert decimal 93 to BCD you get 10010011, which looks like binary. Consequently, you might be tempted to convert this "binary" number to hex, by following the standard procedure, to get 93.

This might fool you into thinking that hex 93 is the same as decimal 93, which is not so. The "hex" 93 is not a true hexadecimal number; it is only a form of shorthand that allows you to express the bit pattern 10010011 in a simpler form. If you were employing an assembler that used hex, you might use what looks like hex 93 when you really meant BCD 10010011.

BCD to Decimal

This conversion is the same as that for binary to hex: Arrange the bits in groups of four starting from the right, and convert each group into hex using the table. For instance, BCD 10001001 is grouped into 1000 and 1001, which gives the decimal 89.

In BCD to decimal, you should *never* get the digits A through F. If you do, then the BCD number was wrong. For instance, to convert 00111100, you would get two groups 0011 and 1100. The 0011 converts into a 3, but 1100 converts to C, which is not allowed in decimal. Hence, 00111100 was not a valid BCD number.

So — What's All This Used For?

If all your programming is in BASIC, you will probably never need to know any of this hex magic. But if you do any machine- or assembly-language programming, it will help a lot.

For example, suppose you want to set up a counter at -50 (decimal) and want to convert this to hex. First find +50 in hex: 50 divided by 16 is 3, with a remainder of 2; 3 divided by 16 is 0, with a remainder of 3. So, a decimal +50 is hex 32. Now change this to -50 by finding the two's complement:

15	16
- 3	- 2
12 (C)	14 (E)

-50 is CE in hex.

Or suppose you want to subtract 2 from some hex number. If your computer does not have a subtract instruction, you can do the same thing by adding a -2. In hex, 2 is 02, and the -2 is found as the two's complement:

15	16
- 0	- 2
15 (F)	14 (E)

You should add hex FE.

Once you figure it out, hex magic can be fun. ■

KILOBAUD KLASSROOM NO. 10

Bus Traffic Control

George Young
Sierra High School
Tollhouse CA 93667

In the last session, we covered the majority of the TTL counters and some of the register chips. We performed many experiments with these chips, thus building your background skills in reading circuit diagrams and, I hope, building up your confidence as well. As you can see, the sessions are beginning to get a bit rougher. Hang in there; we will make it yet.

In this session, we will take up decoders, decoding, three-state devices, and how traffic is controlled on the microprocessor data bus.

Introduction

Most of our modern microprocessor chips have 16 address lines providing the capability of selecting 65,535 discrete memory locations. These separate memory locations are referred to as the address space of the microprocessor. Fig. 1 shows the microprocessor and 16 address lines. These are labeled A_0 through A_{15}. 1K of RAM requires ten address lines from the microprocessor to select the 1024 separate memory cells in each RAM chip; so we have drawn the 10 address lines A_0 through A_9 running from the microprocessor to the 1K RAM block.

We are going to draw the 1K RAM block in an unusual fashion. There are actually eight separate RAM chips in the RAM block, and we have drawn them stacked up in order to conserve space. We did not draw eight rectangles in the stack, but the concept of more than one chip is readily conveyed by this diagram.

As shown in Fig. 1, our 1K of RAM will not function; two things are wrong. First, the ten address lines will not drive the address inputs of the 1K RAM block. Microprocessor output pins are capable of driving one TTL load. We are asking each address line to drive eight inputs to the RAM block. Therefore, we must provide buffering on each of the address lines out of the microprocessor. A buffer is a circuit placed between two circuits to provide isolation. We need a buffer on each address line not for isolation, but to increase the drive capability.

The second reason Fig. 1 won't work is that the chip enable ($\overline{CE}$) pins on the RAM chips are *floating*. The $\overline{CE}$ and the small circle on the symbol both indicate that we need an active low enable here to make the RAM function.

In Fig. 2a, we have added noninverting buffers to each address line to provide the drive capability required. Our first idea is to use the A_{10} address line for the $\overline{CE}$ input for the first 1K RAM block. After all, this line will be low for the first 1K of memory space; and when this line goes high, the first 1K RAM block will be de-selected.

We are also introducing another concept in Fig. 2a. The ten address lines, A_0 through A_9, are shown entering a rectangle. Feeding from the rectangle is a widened arrow that goes to each of the RAM blocks. Data lines and address lines are often drawn in this fashion. The broadened line indicates that more than one line is included in the wide line. This saves drawing the individual lines involved and takes less space in the diagram. As long as the idea is understood by everyone, there is no problem, and the diagram is clearer and actually more easily understood.

Furthermore, in Fig. 2a we have added a second 1K RAM block. Our first thought on

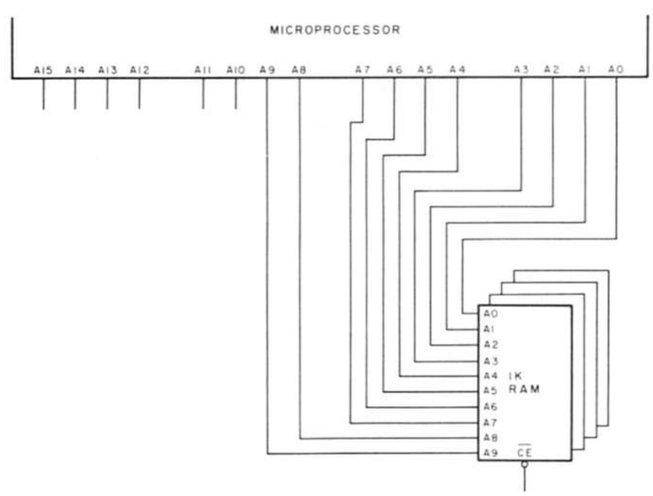
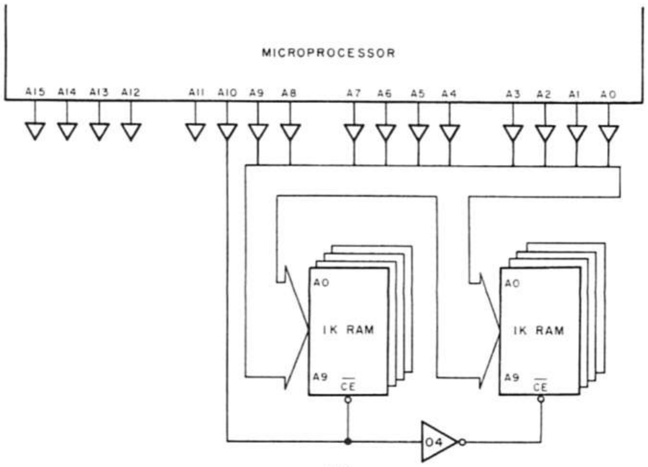


Fig. 1. Addressing the 1K RAM block.



ADDRESSES	A15	A14	A13	A12	A11	A10	
4-1023	0	0	0	0	0	0	1ST 1K
1024-2047	0	0	0	0	0	1	2ND 1K
2048-3071	0	0	0	0	1	0	3RD 1K
3072-4095	0	0	0	0	1	1	4TH 1K
4096-5119	0	0	0	1	0	0	5TH 1K
5120-6143	0	0	0	1	0	1	6TH 1K

(b)

Fig. 2. Adding the second 1K RAM block.

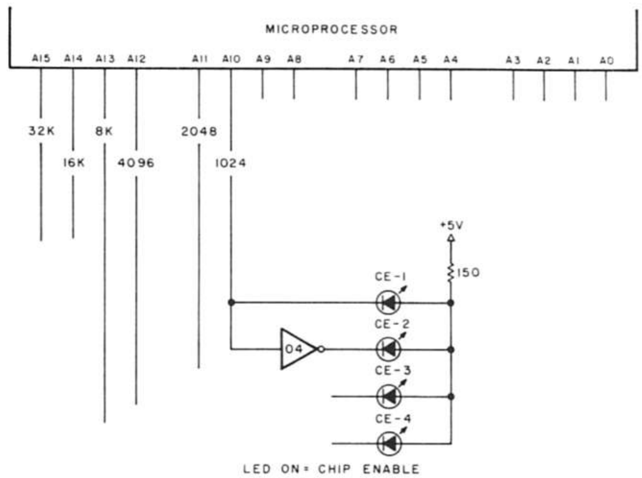


Fig. 3. Experimental setup for decode testing.

handling the second group of $\overline{CE}$ pins on this block is to add an inverter between the $\overline{CE}$ on the first RAM block and the $\overline{CE}$ on the second RAM block. This will work *if* we only have 2K of memory in our system. If we have more RAM or ROM, then an examination of the truth table in Fig. 2b will help us find out why this simple method of enabling the 2K will not work.

The truth table shows that the A_{10} line does indeed start out low for the first 1K of memory space and then is high for the second 1K. But lines 3 and 5 of the table also show the A_{10} line low. Therefore, the first 1K RAM block will be selected every time the A_{10} line goes low. In other words, the single inverter decoder will not do for memory sizes above 2K.

Fig. 3 shows the experimental setup for the design console breadboard and the address lines from the microprocessor. Since we don't have a microprocessor (yet), we'll use this circuit to show how the lines are related; the actual test circuit is shown in Fig. 4a. The chip enable LEDs have been arranged in the circuit to turn on the LED when the CE line goes low.

In Fig. 4 we are attempting to place an equivalent circuit on the console breadboard that will represent what happens with the address lines and the decoding process. Fig. 4a shows the equivalent breadboard circuit for Fig. 2. Note that we are not considering the A_0 through A_9 address lines in the decoding process. These lines are used by each 1K block of memory throughout the address space and are not used in the decoding process for each 1K block.

In order to have a 1K RAM block selected only once in the memory space, we must use some form of decoding. We can use gates and inverters and decode each 1K block in this fashion. Fig. 4b shows this kind of decoder. You can set this circuit up on the console breadboard and use it to decode the four CE lines; but there is an easier way—use a decoder chip. This makes a rather long introduction, but I think that we have the problem fairly well delimited.

Experiment #51
The 7442 Decoder

Problem: How can the address lines of the microprocessor decode the memory chips?

Solution: We will investigate this on the console breadboard. The experiment uses the 7442 decoder, but the 7441, the 7445, the 74145, the 8250 or the 8251 may also be used for this experiment.

Procedure: Refer to Fig. 5. Fig. 5a shows the 7442 pinouts; Fig. 5b shows the 7442 truth table. Notice the row of zeros (lows) traveling diagonally across the truth table . . . this is exactly what we need for chip enable pins. Put the 7442 on the console breadboard (don't forget power and ground). Use four jumper wires to represent the A_{10} through A_{13} address lines. Start with all four inputs to the 7442 grounded. The LED marked CE-1 should be on.

Theory: The 7442 is a one-of-ten (usually written 1:10) decoder. It has four input lines marked A, B, C and D on our diagram. The truth table of Fig. 5b

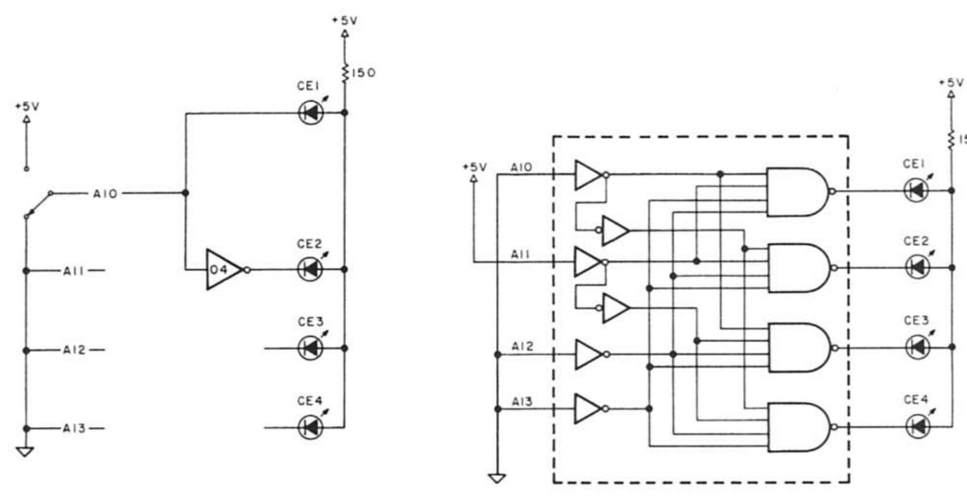


Fig. 4. Delimiting the address decoding problem.

shows that with all inputs low, the 0 output line (pin 1) will be low. This should turn on CE-1. This line would, therefore, go to the first 1K RAM block $\overline{CE}$ pin, and would select that RAM block. (Fig. 5c illustrates decoding 4K of RAM.)

Now take the A_{10} jumper wire high. This should turn on CE-2 and turn off CE-1. This line (from pin 2 on the 7442) would go to the second 1K RAM block and select this RAM block while, at the same time, the first 1K RAM block is de-selected.

If you now encode a binary 2 by taking the A_{11} line high and the A_{10} line low, pin 3 on the 7442 should go low, turning on CE-3 and turning off CE-2. This line from pin 3 on the 7442 would go to the third 1K RAM block and select it while blocks 1 and 2 are de-selected.

Finally, if you encode a binary 3 with both the A_{10} and A_{11} lines high, CE-4 will illuminate and CE-3 will turn off. Pin 4 of the 7442 would go to the fourth 1K RAM block selecting it while the highs on pins 1, 2 and 3 will de-select the first three RAM blocks. Thus, we have a decoder for 4K of memory chips.

But wait, we did not use all the outputs of the 7442. What about the rest of the output pins?

The 7442 may be operated as a 1:4 decoder, 1:8 decoder or 1:10 decoder. To use only the first eight outputs of the 7442, we do not use the D input to the 7442; we leave it grounded. We can then operate the 7442 as a 1:8 decoder and use the eight output pins to decode 8K of RAM. To operate the 7442 as a 1:4 decoder as we just did in the experiment, leave the C and D inputs grounded and operate the 7442 as a 1:4 decoder to decode 4K of address space. We may use all ten out pins of the 7442 and decode 10K of address space with the 7442.

Fig. 6 gives the pin-outs for several more decoder chips.

Experiment #52
The 74154 Decoder Chip

Problem: To decode more than 10K of address space.

Solution: Use a decoder that has more output pins.

Procedure: Refer to Fig. 6e, where the 74154 1:16 decoder is set up in a test circuit. This 24-pin chip was designed for address decoding in computers. It has two enable pins, 18 and 19. Use two jumper wires on these pins to represent the A_{14} and A_{15} address lines. Any binary counter may be used to simulate the A_{10} through A_{13} address lines. Set up the circuit with the 74161 counter chip. Sixteen LEDs are shown monitoring the 74154 output lines.

If you do not have 16 LEDs, then use as many as you can for the test circuit. Remember that the console logic probe may be used for one LED and that you have eight LEDs in the console 7-segment readout. If you have the FND 70 readout, then it will be necessary to drive the segments of the FND 70 through inverter sections since FND 70 requires an active high to turn on each segment. The 74154 will decode 16K of address space.

Experiment #53
The Traffic Cops

Problem: What is all this stuff hung on the data bus lines?

Solution: Let's take a look.

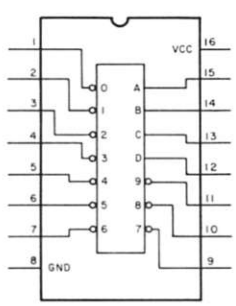
Procedure: Fig. 7 shows the microprocessor chip and its eight data lines. It also shows arrows signifying data traveling both directions on these data lines. During a read cycle, the data is traveling from memory (or input/output devices) into the microprocessor. During a write cycle, data travels from the microprocessor out to external devices. Fig. 7b shows a single data line (D_0) and a pair of open collector NAND gates acting as traffic cops on the data line.

Theory: Assume that the microprocessor is in a memory read cycle. This means that the $R/\overline{W}$ is high. The high on pin 2 of the 7403 will enable this gate, which means the data to be read into the processor will be enabled. This high is also inverted to a low by the inverter section, and the low on pin 5 will disable this gate (taking it out of the circuit for the time being).

Next, the microprocessor is assumed to go into a memory write cycle. The $R/\overline{W}$ goes low, and the low on pin 2 of the 7403 now disables this gate and pin 3 floats on the end of the 2.2k pull-up resistor. The low on the $R/\overline{W}$ line is inverted by the inverter section, and the resulting high output applied to pin 5 will enable this gate. The data to be written into memory (from the processor) will now be enabled onto the data bus. This circuit illustrates how the two-way traffic on the data bus is controlled by the "traffic cops" in the circuit. The $R/\overline{W}$ line and the inverter control the two gates and the direction of the traffic flow.

Fig. 7b is fine for an introduction and example of controlling data on a bus going to and from the processor. However, it isn't practical from a design standpoint (for several reasons). First, the dual-gate configuration would have to be repeated for each data line. This means the $R/\overline{W}$ output from the microprocessor would be driving *eight* gates. You'll recall from an earlier discussion that all of the microprocessor outputs are capable of driving only one TTL gate each. Fig. 7c illustrates a solution to the problem—the addition of an inverter, and a little reconfiguring. Now the $R/\overline{W}$ signal is going into the 7404 (pin 1), which is driving the eight write gates (only one of which is shown).

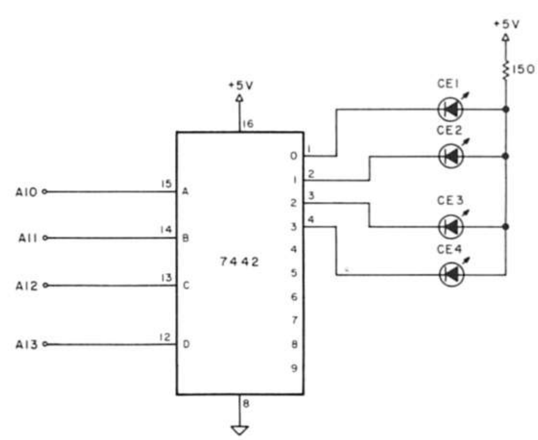
The second, and most important, reason why this circuit is totally unacceptable lies in the use of the 7403 gates for interfacing with the bus. The whole idea behind a bus system is that several devices can be plugged into the bus (i.e., other gates will be tied to the bus further down the line). These additional gates have a "loading effect" on the bus. Without my going into a detailed technical explanation, it will suffice to say that such systems consume a lot of power and are noisy (i.e., have glitches and



(a) 7442 PIN-OUTS

INPUTS				OUTPUTS									
D	C	B	A	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	1	1	1	1	1	1	0	1	1	1
0	1	1	1	1	1	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	1	1	1	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1

(b) 7442 TRUTH TABLE



(c) DECODING 4K OF RAM

Fig. 5. The 7442 decoder.

spikes that can be interpreted as logic ones or zeros). The answer to the problem is to use Tri-state gates for interfacing to a bus.

Tri-state gates, such as the 8T97 shown in Fig. 8a, are either enabled or disabled. When they are enabled by a low on the DISable pins (1 and 15), the outputs will be determined by the logic levels (HI or LO) at the input pins. In other words, the gates are working just like any other gates. When they are disabled (by a high on the DISable line) the gates are effectively disconnected from the bus. The outputs are said to have gone into a high-impedance or open condition and do not present any loading to the bus (i.e., they are disconnected). Fig. 8b is a truth table for the operation of the 8T97 and Fig. 8c illustrates a typical bus interface configuration.

In summary, there are three advantages to using Tri-state gates when you are interfacing to a microcomputer bus (one of which I haven't mentioned before). First, lower power consumption; second, less loading on the bus (thereby maintaining waveform integrity); and finally, higher speed (faster switching from a high to low or vice versa).

Note that the 8T97 is a noninverting buffer and has four sections controlled by one line and two sections controlled by a second line. The two sections may be operated independently of each other. The DM 8097 and the 74367 are also the same type of chip. The 8T97 is more

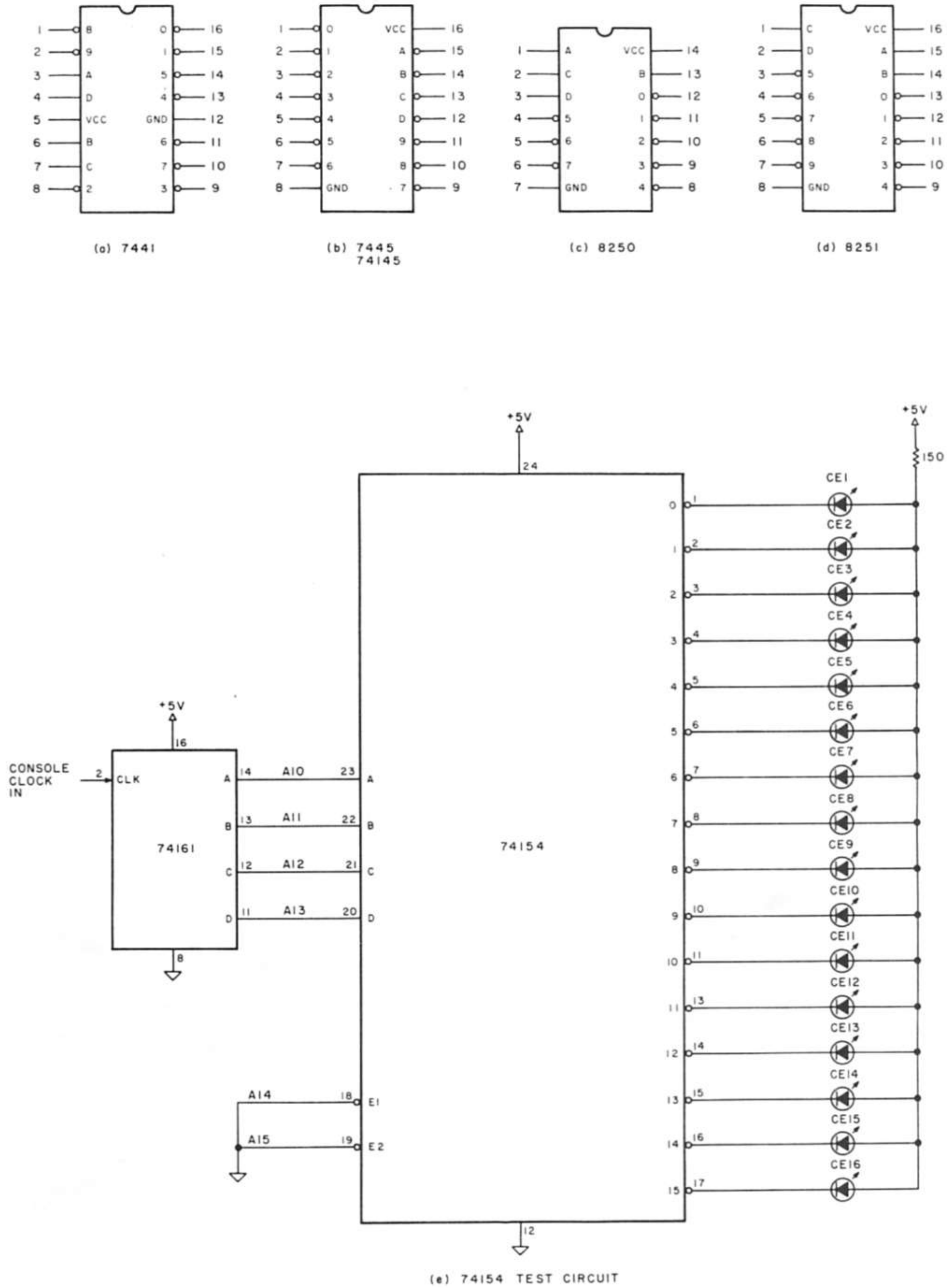


Fig. 6. The 74154 decoder.

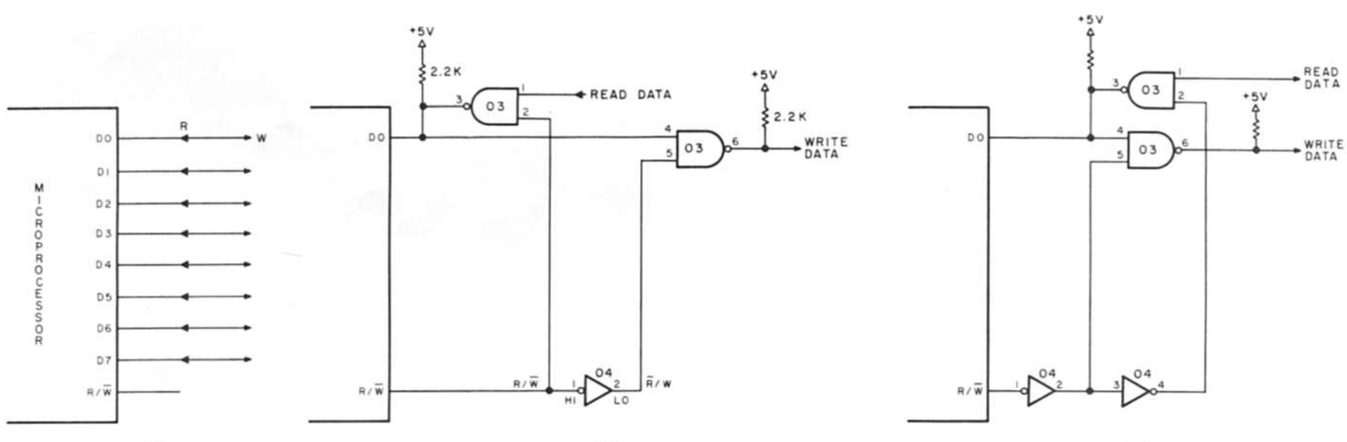


Fig. 7. Traffic control on the data bus.

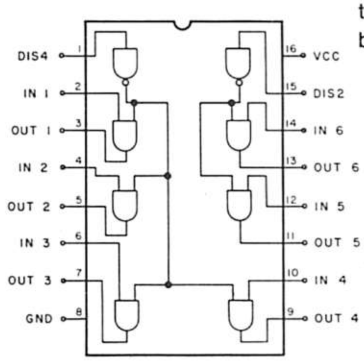
expensive than the others, but my own experience with these chips indicates that the 8T97 has more drive capabilities and proves superior in operation in the circuit . . . justifying its greater cost.

Other chips are becoming available for this buffering job on the data and address buses; I think that soon we may see a new family of microprocessors with the buffers, as well as RAM and ROM, built into the chip. In fact, Intel has a new microprocessor chip, with many of these capabilities built in, which will be second-sourced by Signetics. This points the way that things are heading in the subsequent generation of microprocessor chips.

Preview

We have looked at the microprocessor address bus, how decoding of the address space may be accomplished and how traffic is controlled on the data bus.

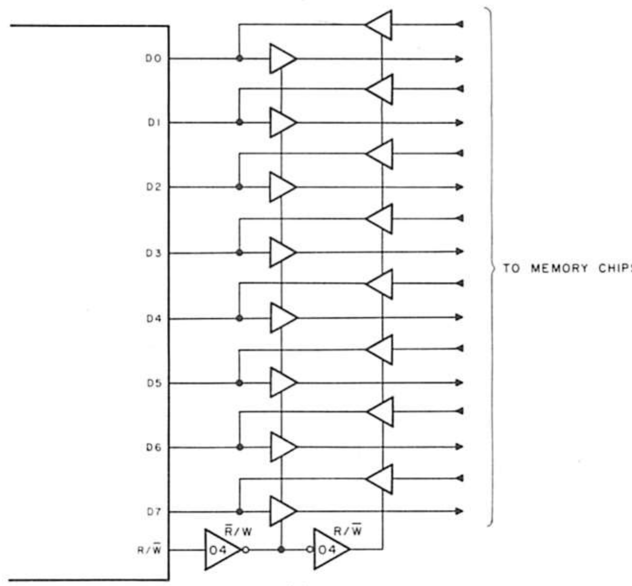
Next time we will turn our attention to the memory chips, both ROM and RAM. Using the 7489 (8225), we will set up 64 bytes of memory on the console breadboard, and also burn a 7488 (8223) PROM on the console. Sierra Electronics, Box 11, Auberry CA 93602, will furnish a package for us of two 8225s and two 8223s for $4 postpaid in the U.S. and Canada. California residents, add 6 percent sales tax. ■



(a) 8T97 PIN-OUTS DM8097 74367

DISABLE DIS2	DIS4	INPUT	OUTPUT
0	0	0	0
0	0	1	1
0	1	X	HI Z
1	0	X	HI Z
1	1	X	HI Z

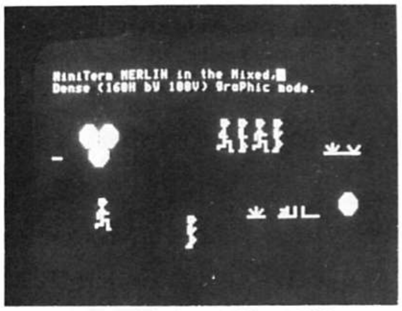
(b) TRUTH TABLE



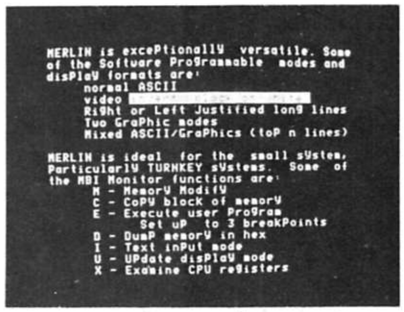
(c)

Fig. 8. 8T97 data bus buffering.

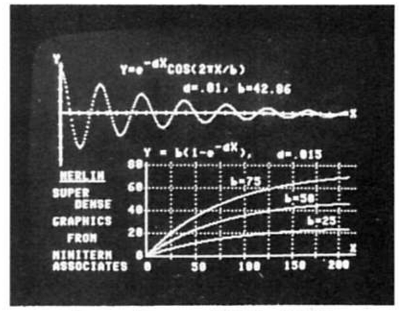
The many faces of MERLIN



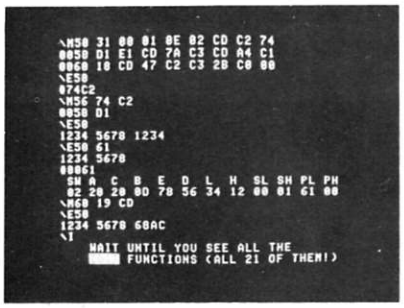
Dense Mode: 160H × 100V
Running Man Patterns



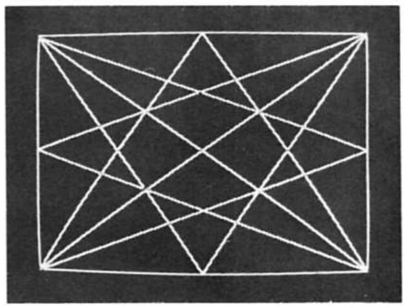
Propaganda



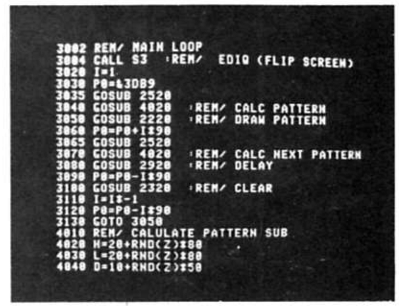
Super Dense: 320H × 200V
Equation Plotting



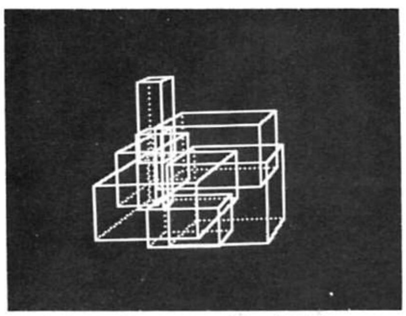
Monitor Debug Usage



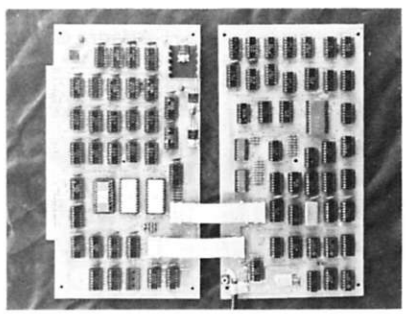
Super Dense: 320H × 200V
Line Drawing

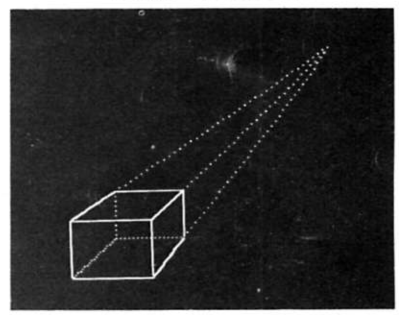


BASIC Program Listing
Output Shown Below



Super Dense: 320H × 200V
3-D Boxes





Super Dense: 320H × 200V
Perspective Drawing

MERLIN (and your S-100 Computer...)
... the graphics development package you've been waiting for.

Do these photos suggest an application in your field? Whether you're into architecture, astrology, music or whatever, you can apply graphics to make your presentations more effective and your work more efficient.

Real Time Plotting
- Heart Rate
- Navigation
- Spectrum Analysis

Complex Equation Plotting
- Stock Market Trends
- Teaching: Mathematics
- Circuit Responses

Fine Line Drawings
- Architecture
- 3-D Projections
- Circuit Layouts

Pattern Movement
- Animation
- Games

MERLIN is also a multipurpose system monitor board with ROM monitor and editor software, parallel keyboard port, and audio cassetts storage, besides being your text and graphics video output device.

Write or call today for your free copy of our new catalog, an index to available application notes, and a list of **MiniTerm** Dealers.

MERLIN is supported by expert technical assistance, a constantly expanding series of application notes, and a newly formed users group.

Assembled and tested **MERLINs** start at $349, $269 for kits. Super Dense add-on, firmware, and cassette interface are extra.

MiniTerm Associates, inc.

Dundee Park, Andover, MA 01810 (617) 470-0525

M40

Expand Your KIM

Part 5: A/D interfacing (for joysticks!)

John Blankenship
datamart, inc.
3001 No. Fulton Dr. N.E.
Atlanta GA 30305

No matter what kind of computer you have, this article can help you add four channels of analog input for a fraction of the cost of other methods I've seen. If you've been building the KIM System, this analog board will complete the project.

I designed the KIM System with several requirements in mind for the analog ports: I required four channels (so that two joysticks could be inter-

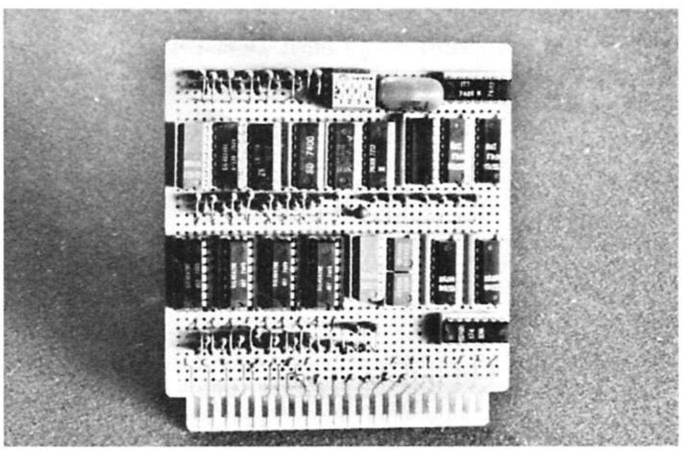


Photo 1. Four channels of A/D, two channels of D/A and an input port for sense switches.

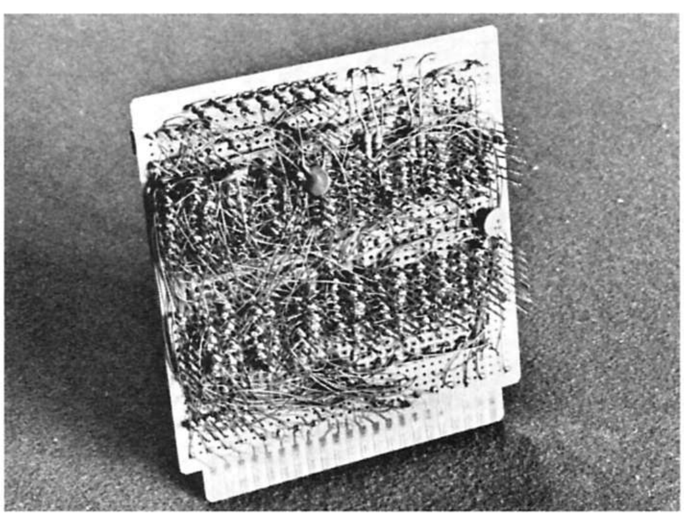


Photo 2. Circuits are wire-wrapped on a 44-pin board.

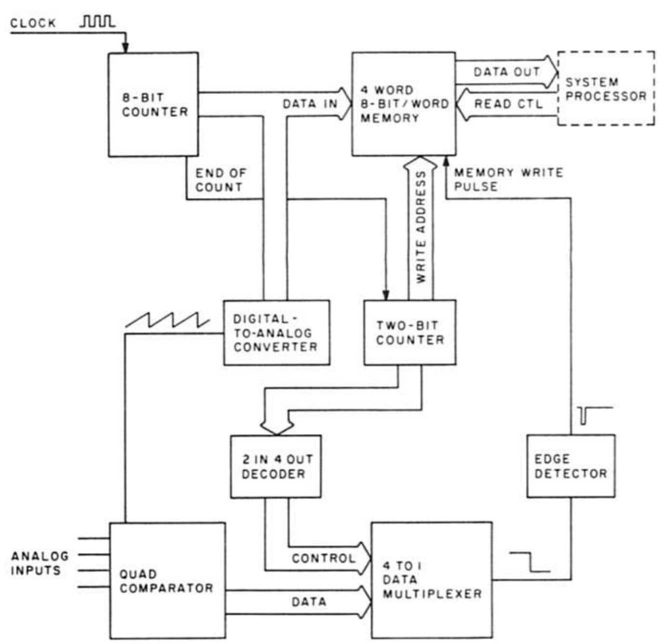


Fig. 1. Block diagram of the A/D converter.

faced), with each sampled often enough to provide reasonable accuracy for use as a video game input device. To make the use of these ports easy, I wanted each to be read as a normal memory. Finally, each of the A/D (analog-to-digital) channels had to be easily switchable to other devices besides the joysticks.

Besides the A/D ports, I also wanted at least two D/A (digital-to-analog) ports to experiment with music, speech synthesis, motor control, etc. I also wanted a port for sense switches to give me a full complement of methods for interfacing with my machine. I combined all these circuits on one board and labeled it *External Interfacing* in my previous articles.

Photos 1 and 2 show the board itself. Although I was able to cram the circuit onto a 4½-inch-square board, I would recommend epoxying a vector board on the top to give more room for the components.

Fig. 1 shows the basic block diagram for the A/D circuits. The four-word memory is one of the major secrets of making this circuitry both inexpensive and easy to use. This memory is made up of two 74LS170 chips composed of four 4-bit words each. I chose these chips because they have separate *read* and *write* controls, thus enabling read and write operations to occur simultaneously.

The A/D circuitry will update each of these memory locations with a number that is proportional to the analog input. The output of the memory chips is connected to the data bus so that they appear as standard memory to the processor.

The eight-bit counter continually generates sequential numbers from 0 to 255. A D/A converter converts these numbers to an analog voltage which, for all practical purposes, is an increasing ramp. This ramp is fed to four comparator circuits that compare the ramp voltage to the analog inputs.

The comparators output a level 1 when the ramp voltage equals the analog input. Since the ramp voltage also equals the number in the eight-bit counter, it is implied that the instant a comparator fires, the eight-bit counter contains the digital equivalent of the analog voltage being applied to that comparator.

The remainder of the circuit has one major function . . . it must decide which comparator fired, and form an address for the four-word memory so the eight-bit counter data can be gated into the appropriate location.

I chose to control the write address with a two-bit counter. Since this counter increments every time the eight-bit counter completes a full cycle, the addresses 0, 1, 2 and 3 are being applied sequentially to the *write* address, and each is held there for the full cycle of the eight-bit counter.

Additionally, this two-bit counter is decoded and used to enable only one of the four comparators (the one corresponding to the write address) at a time. The level change indication from the multiplexer is converted to a narrow pulse and used to activate the write line on the memory chips.

As explained above, the four memory locations are continually, and automatically, refreshed with the digital equivalent of four analog inputs. The processor needs only to read these locations for the latest updates.

Fig. 2 shows the actual schematic of the A/D circuit. The 7493 simply reduces the frequency to a trackable rate. The 1408L8, D/A converter, outputs a current ramp that is converted to a voltage ramp by the 741

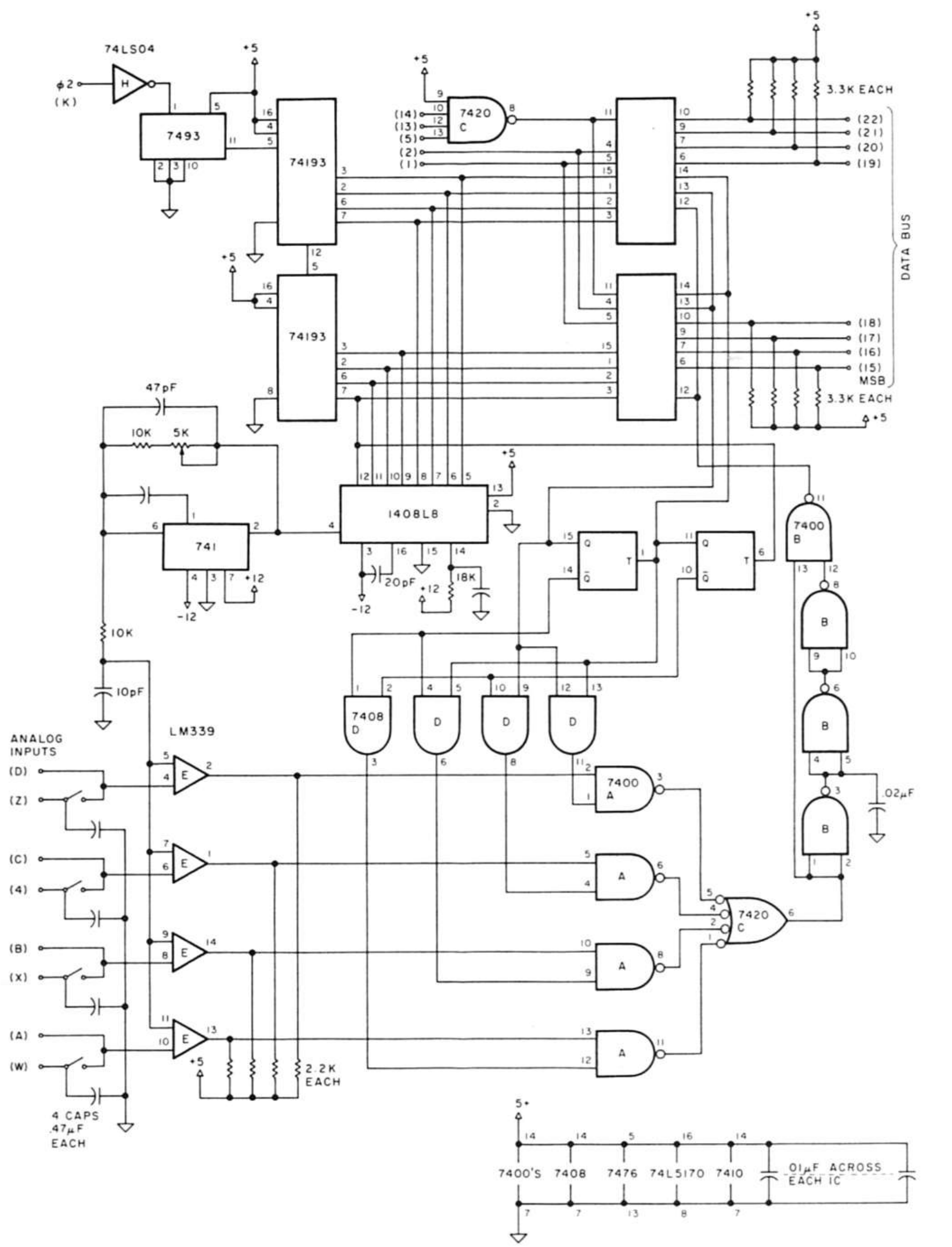


Fig. 2. Schematic of A/D converters.

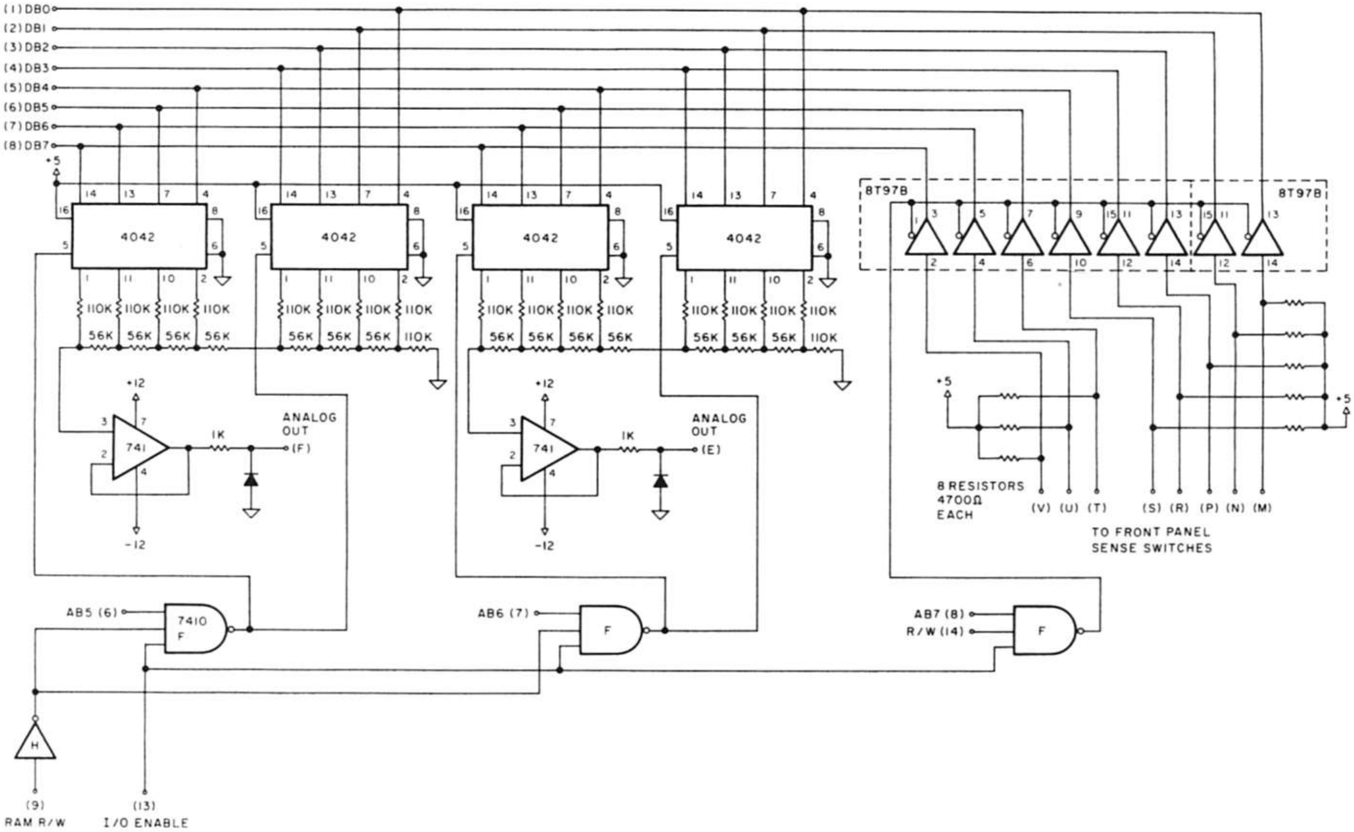


Fig. 3. Schematic of D/A converters and input port for sense switches.

op amp. The 7400 labeled B acts as a one-shot to perform as the edge detector.

Half of the 7420 is used to decode the address bus for processor reads. Address decoding will be discussed in more detail later in this article.

Since the 74LS170s are open collector, rather than Tri-state outputs, pull-up resistors are required for interfacing with the bus. The DIP switch disconnects the joystick inputs. Once they're disconnected, you can input other signals to the converter by way of the backplane jacks (see my earlier articles).

The other two functions, D/A and sense switches, are detailed in Fig. 3. Since I felt that the accuracy of the D/A conversion was not critical, I chose not to use the Motorola D/A converter chip used in the A/D circuit. If I had used the Motorola chip, I would have had to use two eight-bit registers to hold the data, the two D/A chips themselves and a current-to-voltage converter.

I chose to use MOS registers for my output ports. Since MOS gates output *exactly* Vcc and zero volts for their corresponding high and low levels, I used them to drive a resistive ladder directly. Additionally, since MOS chips represent a very small load, they can be hung on the bus without buffering. (Note: MOS chips do represent a relatively large capacitive load, and hanging them directly on the bus is not good practice in expandable systems. In this case, however, I knew exactly what loads I would be dealing with and was able to determine that enough drive capability was present.)

The 741s in Fig. 3 are used as unit gain amplifiers for buffering purposes. The 7410 is used to decode out the address lines to determine which port is being used. The sense switches are connected to the inputs of Tri-state buffers. The outputs of these buffers gate the switch data onto the bus when enabled.

Port Function	Page	Loc
Dazzler Mode control	80	0F
Dazzler ON/OFF, Address	80	0E
Right vertical joystick	80	10
Right horizontal joystick	80	11
Left vertical joystick	80	12
Left horizontal joystick	80	13
Sense switches	80	80
D/A port A	80	20
D/A port B	80	40

Fig. 4. Summary of special addresses used by the KIM-1 System.

A/D D	A	1	AB0
A/D C	B	2	AB1
A/D B	C	3	AB2
A/D A	D	4	AB3
D/A B	E	5	AB4
D/A A	F	6	AB5
−12	H	7	AB6
JS REF. Volt.	J	8	AB7
Ø2	K	9	RAM R/W
Ground	L	10	+12
SS 0	M	11	
SS 1	N	12	+5
SS 2	P	13	I/O ENABLE
SS 3	R	14	W/R
SS 4	S	15	DB7
SS 5	T	16	DB6
SS 6	U	17	DB5
SS 7	V	18	DB4
JS LH	W	19	DB3
JS LV	X	20	DB2
JS RH	Y	21	DB1
JS RV	Z	22	DB0

Fig. 5. Pin-out designations for the external interface board.

In order to better understand the I/O functions, you might reread my article ("Expand Your KIM!" Part 3, *Kilobaud*, February 1978, p. 68) in which I explain how I decoded part of the address lines to indicate an I/O operation, rather than a memory transfer.

All my I/O ports (including the four-word memory used for A/D) are partially enabled by this I/O enable. Since I know how many total ports I designed for, I only partially decoded the low-order address lines. This drastically limited the number of ports available on the KIM System, but the ease of implementation, as well as the reduction in cost, made it well worthwhile.

Fig. 4 summarizes the I/O addresses used uniquely by my system. If you convert these hex addresses to binary, you can see how the appropriate address lines are used to enable each port decoder.

There are only two major differences between input and output decoding. The first is that the R/W or the $\overline{R/W}$ line is used to indicate the direction of the transfer. Second, the *write* pulse for an output port must be coincident with the trailing edge of the Ø2 clock. Again, I refer you to Part 3 of this series for more details.

Fig. 5 shows the pin-out designations for the external interfacing board. These match the mainframe wiring done in Part 2 of this series.

In order to insure that builders of the KIM System fully understand how to utilize the joystick interface, I have included a short program in Fig. 5 that will enable you to draw with the joystick on the TV screen. The sense switches control the colors of the two-color dot that is moved by the joystick.

This program serves a useful function as an educational endeavor, and that's about all. However, I do feel that builders of the KIM System will find it useful as a reference. I have tried to functionally describe each section with comments.

This completes the hardware series on my KIM-1 system, which now contains 17K of RAM and supports both BASIC and FOCAL. I'm also in the process of implementing a new language with an ease of use and a speed of operation somewhere between assembly language and BASIC.

Because my system is to be multilingual, I have chosen to avoid ROM in favor of RAM for all functions except the KIM monitor. I'm also planning several surprises that I hope to share in the future. ■

Address	Contents	Label	Mnemonic
00 00	LOC PAGE	DATA	STORE POINTER
	:Set mode and starting address for the dazzler		
02	A9 10	INIT	LDA #$10
04	8D 0F 80		STA MODE
07	A9 90		LDA #$90
09	8D 0E 80		STA BEGADDR
	:Get horizontal joystick position		
0C	AD 11 80	START	LDA JOYHOR
	:Place 4 MSB into 4 LSB and save		
0F	4A		LSR
10	4A		LSR
11	4A		LSR
12	4A		LSR
13	85 00		STA LOC
	:Get vertical joystick position		
15	AD 10 80		LDA JOYVER
	:Check for and set up proper page of screen display		
18	30 07	TOP	BMI BOTTOM
1A	A0 20		LDY #$20
1C	84 01		STY PAGE
1E	4C 25 00		JMP CONT
21	A0 21	BOTTOM	LDY #$21
23	84 01		STY PAGE
	:Remove MSB and keep only the next four		
25	0A	CONT	ASL
26	29 F0		AND #$F0
	:Combine LSB and MSB into one word and save		
28	05 00		ORA LOC
2A	85 00		STA LOC
	:Put color (sense switches) into accumulator		
2C	AD 80 80		LDA SENSE
	:Prepare for an indirect store using 00 and 01 as pointer		
2F	A2 00		LDX #$00
	:Store color		
31	81 00		STA LOC PAGE
	:Begin Again		
33	4C 0C 00		JMP START

Fig. 6. Sample program for drawing on TV using joystick.

Sheila Clarke
CyberGrafix
518 N. Brand
Glendale CA 91203

What's Happening with the IBM Selectric?

Micro Computer Devices has the answer

Art Childs needed a printer for a long time. An IBM Selectric had been Art's and my choice for a couple of years. It was ideal because of its small size and beautiful print quality. However, we both were skeptical about printers available for use with a computer. In most cases, either the typewriter was used and reconditioned or a lot of interface kit assembly was required. Like many computer-users, Art can't afford to risk having an unreliable unit requiring continual maintenance; nor does he have time to assemble a kit.

When we first heard about the SELECTERM, made by Micro Computer Devices (MCD), we were impressed that someone had finally converted a brand new typewriter for microcomputers. Because both IBM and Micro Computer Devices provide warranties for their respective portions of the device, we decided to obtain a SELECTERM.

Art's system consists of an Altair 8800, dual ICOM floppy-disk drive, and an ADM CRT terminal. He uses the 3P + S interface board from Processor Technology. After spending two hours struggling to decipher the board's schematics, which seemed to be written solely for hardware types, Art finally called his engineer friend, Steve Griffis, who came over and had everything running in five minutes. Although the SELECTERM will interface to any microcomputer, what if you can't read the interface board schematics? Micro Computer Devices is providing a solution with specific connecting instructions for each interface available on every computer now being sold.

Art's reaction to the printer was positive from the moment the two large cartons were delivered. One carton held the Selectric and the other contained the electronics package. He was impressed with the packing, which held the units solidly with formed foam to prevent damage caused in shipping. Opening the flap of the carton, Art uncovered a sheet that said STOP, with complete unpacking and typewriter assembly instructions. Art, in too big a hurry, merely made a mental note that instructions were there and, consequently, ran into a little trouble securing the cover latches of the typewriter. (Sometimes I wonder if anyone reads anything before making panic calls to the manufacturer.)

He was also impressed with the documentation and the SELECTERM's acceptance of ASCII. With no conversion necessary, Art began writing a driver. It took him five minutes, using assembly language for FDOS-III. He said the only difference between this printer and another line printer driver or hard-copy output driver is that you might have to put out some nulls after tabs and line feed. But it was simple for him to write the nulls into the driver. Fig. 1 shows the driver for the 8080 and 3P + S.

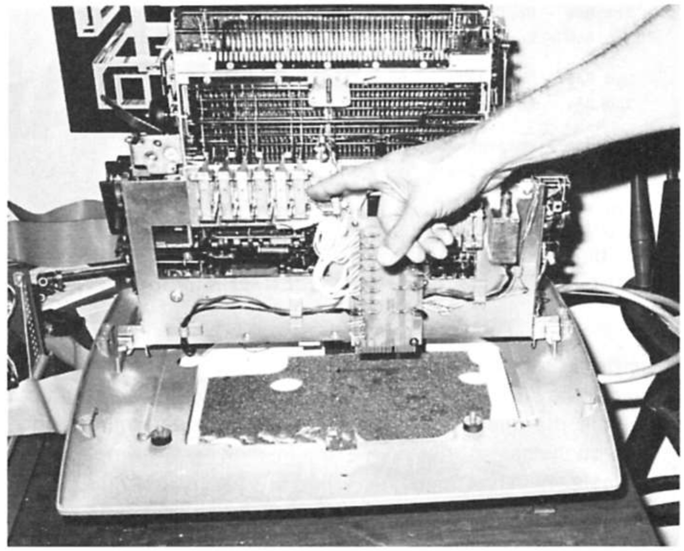
Art commented: "The IBM print quality is nice. And I like the fact that I can change type fonts. Putting the whole thing together—removing it from the cartons to putting the cover on the typewriter and hooking up the cables—was a half-hour task. The fact that it requires one parallel port makes it easy. If you have only one serial port, which is often the case, you'll usually lose it to your print device. Writing the driver and integrating it into the software completed the process. All in all, it was very easy; everything's been done for you. The unit runs very cool, the electronics box is barely warm to the touch after running consistently for about three hours, and it runs cooler than the typewriter itself."

I'm using the SELECTERM to prepare this article for *Kilobaud*; I am inputting the text in the computer, from first draft to the final, edited version. It's a pleasure to know I don't have to retype this thing two or three times before I get it right. The advantages of the SELECTERM are only evident when I begin to use it. For example, the sales literature doesn't tell me how to input uppercase and lowercase letters with a terminal that has only uppercase. So MCD owner Shelly Howard pointed out the ADM has switches beneath the nameplate. Setting the LC EN switch enables me to input uppercase and lowercase for printer output. I did discover, however, that the switch must always be returned to the UC position after using the Insert mode of the text editor. After that little switch is flipped, the CRT may only see uppercase characters, but when I hit shift for uppercase characters, the printer outputs caps where they should be—just like using a typewriter.

After using the SELECTERM for a couple of weeks, Art and I ran into difficulty getting clear print—then it jammed. The problem was a loose motor mount. Because the typewriter portion was under warranty, IBM service came out and fixed it at no charge.

How It All Began

To find out how his product came about I spent some time talking with Shelly Howard. Like many other small-scale manufacturers, Shelly knew relatively little about microcomputers two and a half years ago. In fact, he was preparing his thesis for his PhD on an IBM Selectric. After gathering sufficient research data, he wanted it compiled through a computer and output on a Selectric that matched the type of his own typewriter. He was told by two computer outfits that IBM had discontinued making its I/O device. He was forced to either scrap his original plans or buy his own computer. Assuming the cost of ownership would be prohibitive, he searched and discovered the world of microcomputers. He also discovered Don Lancaster's *TV Typewriter Cookbook*.



If we lift the typewriter up off the baseplate, we see the electronics added to convert the typewriter to a printer.

Now They Tell Me!

Although he followed the book's instructions to the letter, Shelly failed to get a unit up and running. He later discovered the book had been based on theory only; no one in Lancaster's organization had actually put the theory to practice. By now Shelly was too committed to back out, so he decided to start over with the help of two design engineers, Steve Garner and Jimmy Carter (no, another one).

Months of design development, field testing and improvements resulted in production of a printer with all parts—the baseplate, actuators, coils, transformer and linkages—manufactured by MCD. Finally, the design was approved by IBM. That's why IBM service will come and fix your printer if anything goes wrong; you can also buy yearly service agreements from IBM after the warranty expires. For this reason, MCD will not sell the SELECTERM in kit form. IBM has only approved the factory assembled and tested model.

In Full Swing

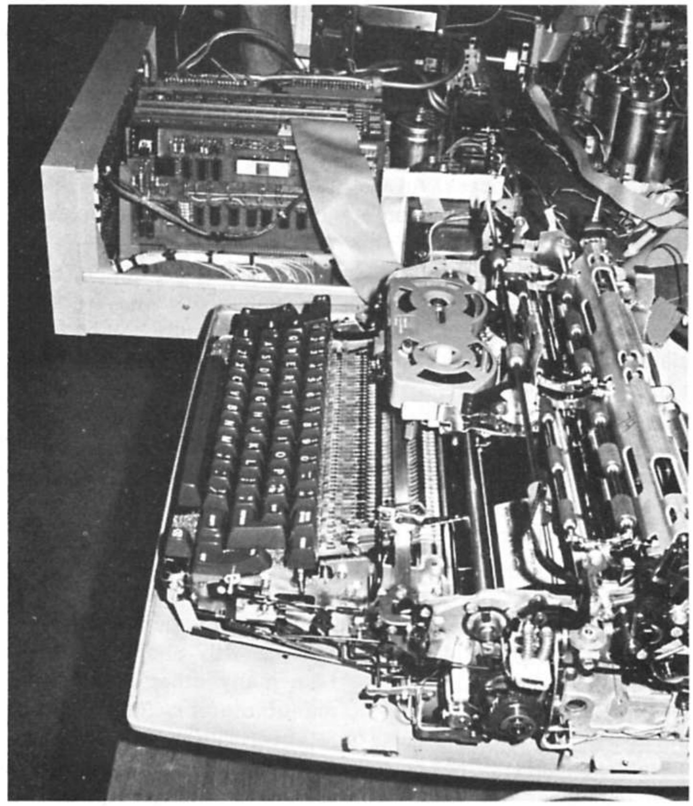
First shipments of the SELECTERM were made in August 1977; currently about three per day are delivered to dealers. The target is five per day, but the cash-flow situation is tough with MCD in a continual fiscal squeeze. Though IBM sanctioned the design, MCD is treated like any other individual consumer, as far as open credit goes. When you buy in quantity, with no quantity discount, at the same price I paid for my

```
0000                ;        ROUTINE TO DRIVE SELECTERM WITH 8080 AND 3P+S
0000                ;
0000  DB04          LO:      IN     4         ;GET STATUS
0002  E601                   ANI    1         ;MASK
0004  CA0000                 JZ     LO        ;NOT READY
0007  79                     MOV    A,C       ;GET CHAR
0008  D306                   OUT    6         ;OUTPUT
000A  FE09                   CPI    9         ;WAS IT A TAB?
000C  CA1C00                 JZ     LOTAB     ;YES
000F  FE08                   CPI    8         ;NO - BACKSPACE?
0011  CA1700                 JZ     LOLF      ;YES
0014  FE0A                   CPI    0AH       :NO - LINE FEED?
0016  C0                     RNZ              ;NO - RETURN
0017                ;
0017  0E00          LOLF:    MVI    C,0       ;OUTPUT A NULL
0019  C30000                 JMP    LO        ;AND RETURN
001C                ;
001C  C5            LOTAB:   PUSH   B
001D  010004                 LXI    B,400H    ;4 NULLS
0020  CD0000        LOTB1:   CALL   LO        ;OUTPUT
0023  05                     DCR    B         ;LAST ONE?
0024  C22000                 JNZ    LOTB1     ;NO
0027  C1                     POP    B         ;YES - RESTORE B
0028  C9                     RET              ;AND RETURN
0029                ;
0029  0000                   END
TOTAL ERRORS=00
```

Fig. 1. ICOM 8080/Z-80 Reloc-Macro Assembler Ver. 1.0.

1st Row - Uppercase:	!	@	#	$	%	¢	&	*	(	)	_	+
1st Row - Lowercase:	1	2	3	4	5	6	7	8	9	0	-	=
2nd Row - Uppercase:	Q	W	E	R	T	Y	U	I	O	P	¼	
2nd Row - Lowercase:	q	w	e	r	t	y	u	i	o	p	½	
3rd Row - Uppercase:	A	S	D	F	G	H	J	K	L	:	"	
3rd Row - Lowercase:	a	s	d	f	g	h	j	k	l	;	'	
4th Row - Uppercase:	Z	X	C	V	B	N	M	,	.	?		
4th Row - Lowercase:	z	x	c	v	b	n	m	,	.	/		

Fig. 2. ASCII character set for SELECTERM output device.



When the typewriter cover is off, the SELECTERM looks about like another Selectric II. Here it sits alongside Art's Altair 8800 with cabling interface to the 3P + S.

Selectric II, a lot of bucks are going out the door at one time. To handle the dilemma, MCD sells through dealers only, on a COD basis. Because requests have been made by some manufacturers, the firm wants to produce OEM versions to specification. Shelly will probably find investors, or perhaps release MCD for acquisition by another company. But he loves what he's doing: selling and delivering SELECTERMs to dealers across the country.

Competition

Presently, only one other company in the country sells an IBM Selectric printer with ASCII encoding. Other companies offer used Selectrics complete with interfacing. Even reconditioned units will not qualify for the IBM Service Agreement.

If you're looking for a good printer, this could be it. But take heed that 15 characters per second may not be fast enough. Long listings could take hours. For most home computerists, however, speed may not be a determining factor in making a printer selection. And the benefits are numerous: All the basic features of the printer include the special typing element, tab command, back space, vertical tab, bell, serial and parallel interfacing, cable sets and software in PROM within the electronics. Also included is a special ASCII typing element that IBM has produced to MCD specifications. Fig. 2 shows an output of the character set.

The price of $1750 appears prohibitive, until you consider that you'll be using an extremely well-designed unit that will last for years—type fonts are changed at will, no special paper is needed, IBM ribbon is easy to order, and service is virtually hassle-free.

Options

The same extras as those offered by IBM, including dual pitch and correcting feature, can be ordered for your SELECTERM. MCD has developed a noise-reduction feature (recommended if you live in a residential neignborhood).

Tractor-feed platen and RS-232 interface are also being offered as options.

After using the SELECTERM a great deal for two months now, Art and I are definitely convinced that we did a good thing for his computer. And a nice plus is that we now have a second typewriter—that is, when it's not being used with the computer. ■

The Top-Down Approach

with some practical examples

Dr. Lance A. Leventhal
PO Box 1258
Rancho Santa Fe CA 92067

In *Kilobaud* No. 14 ("Why Structured Programming?" p. 84), I discussed structured programming, a method for making the logic of large programs simple and repetitive, thereby making them easier to debug and test. But a further problem in writing large programs is how to put sections of the programs together. This article describes a widely used method called top-down design, by which the programmer starts with an overall outline of the program and proceeds to steadily describe each section in greater detail, debugging and testing along the way in an integrated manner.

Modular Programming

Obviously, a large program can only be written by dividing it into sections. No one (I hope) would simply write the entire program and then see if it worked. Clearly, a better idea is to write a small section, see if that works, correct it, write another small section, and so on. This procedure is known as *modular programming* and the sections of the program are called *modules*.

Some typical modules in an overall accounting, game, word-processing or instructional program might be: I/O routines, file-handling routines, mathematical calculations, string-handling routines, table searches, sorting routines, table lookup and list processing.

The advantages of modular programming are clear.

1. You can check the modules individually and be sure they work properly. Thus, you can assume that any errors in the overall program are in the connections or the supervisor program.

2. You can build a library of modules that will be useful in other programs. Many of the previously mentioned modules will be needed frequently.

3. You can use modules that you have previously developed, found in books or magazine articles, or borrowed from friends. You can also use modules such as file handlers, code converters and I/O handlers that comprise part of your monitor or operating system.

4. You can plan program development and have a reasonable idea of how much progress you have made and what the major stumbling blocks are.

5. You can eliminate many simple errors at an early stage.

Modular programming has serious disadvantages, though. Somehow, the modules never quite seem to fit together at the end. Different modules may use different registers, memory locations or subroutines. Some may wipe out results that others need or not use data that others provide. Module integration often turns out to be a big task you must struggle with after everything seems to be done.

The problem of integrating modules is independent of the problem of testing and debugging them. The modules may all work separately, but still not work together. The catch is that the original debugging and testing checks the workings of the module out of context (i.e., all by itself rather than as part of a complete program).

In fact, debugging and testing a module in isolation can be quite difficult. A game program, for example, may consist of the following modules: (1) determine initial conditions, (2) read and check proposed move (see if it is valid), (3) determine new conditions, (4) print status.

But how can you write the routine that reads and checks the proposed move unless you know the previous state of the game and can see the new state? How will you be able to tell if the MOVE module is working properly? Typically, you will have to either manually enter the required data and examine the results or write special programs to perform those tasks. These special programs (sometimes called *driver programs*) can save a lot of manual effort; however, they introduce extra work and may act quite differently from the real routines for which they substitute. (Note that you don't save the driver programs; you throw them away when the job is done.)

Clearly, the problem of combining modules is even more serious in large commercial programming projects. Not only can the number of modules in a project be very large, but also many programmers may be involved in writing them. Now the problem is to integrate modules written by people with different styles, different levels of expertise, different documentation methods and different interpretations of tasks.

Top-down Design

Most commercial programming shops now use some version of top-down design. This method differs from the more traditional bottom-up design (see Fig. 1) in which the specific modules are written before they are integrated into more complex programs. Top-down design (see Fig. 2) proceeds as follows:

1. The overall supervisor program is written, debugged and tested. Major subprograms are replaced by *program stubs* that may produce the answer to a selected problem, record the entry or do nothing at all.

2. Each stub is then similarly

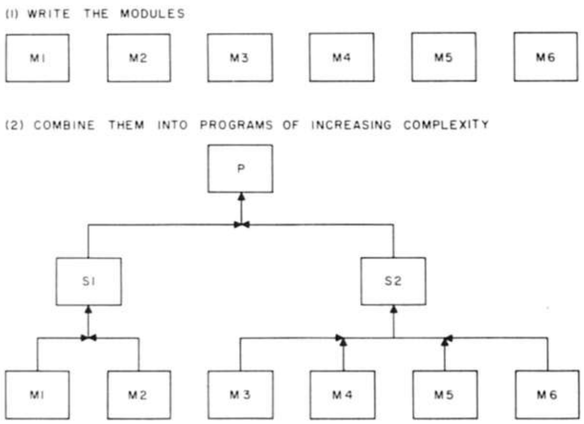


Fig. 1. The procedure for bottom-up design.

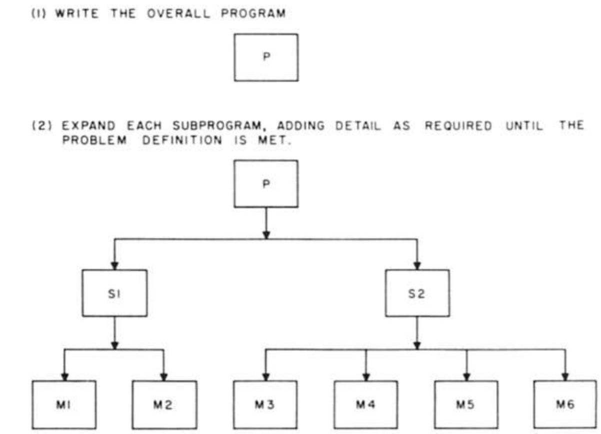


Fig. 2. The procedure for top-down design.

expanded, with debugging and testing occurring at each step.

Advantages of Top-down Design

The advantages of top-down design are:

- It modularizes debugging, testing and integration, as well as coding (the writing of instructions).
- It allows subprograms to be debugged and tested in the actual environment of the entire program. No special debugging and testing programs (or drivers) are needed to provide data or to interpret results.
- It results in overall program logic being checked first. This often means that the programmer can immediately discover and eliminate inconsistencies and misconceptions that otherwise may be very difficult to find and correct (after all the modules have been written).
- It provides a systematic framework for program development and testing. It gives the programmer a firm idea of how much of the task has been accomplished.

Disadvantages of Top-down Design

Of course, like all methods, top-down design has disadvantages. Among these are:

- A suitable program stub may be difficult to write, particularly if it must appear in many different places and produce many different incidental effects.
- The top-down expansion may not mesh well with hardware or already existing software.
- Errors in the overall program can have catastrophic effects on the entire project. Often critical design decisions must be made early before you know what problems exist (or will be created) at the lower levels.

Furthermore, top-down design assumes a simple program structure with independent subsections (i.e., a tree structure, as shown in Figs. 1 and 2). Some programs (perhaps even most) can logically be constructed in that manner. But there is no proof that all, or even most, programs can be. Often programs have interconnections at all levels that defy simple analysis.

Of course, top-down design is no panacea; it provides neither rules nor guidance for: (1) dividing programs into modules that can be written independently of other modules; (2) writing the modules (here, structured programming comes into play); (3) defining or using data structures . . . in many situations, the structure of the data may be more important and more difficult to determine than the structure of the program.

But top-down design does provide a systematic framework, rather than a haphazard approach. This framework has been shown to significantly increase programmer productivity in the commercial world. Furthermore, it seems to result in programs that have clearer logic and are easier to test, debug, extend and use. Of course, programmers should never disdain a little bottom-up design where that method permits better utilization of hardware, existing software or other resources. The aim of programming is to produce programs that work, not to follow the tenets of one methodology or another.

Much of what we have said so far about top-down design is vague. Now let us see how it works in a real example.

The Vote Analysis Program

The purpose of this program is to count ballots and print the totals in decreasing order—starting with the candidates who received the most votes. C is the number of candidates, and the ballots are coded as follows:

0—a blank ballot (no vote for any candidate).
1 to C—vote for the indicated candidate.
C + 1—vote for a write-in candidate.
C + 2—illegal vote (two or more candidates marked).
C + 3—special marking for last (dummy) ballot.

Fig. 3 shows the initial program flowchart. The important variables are: N (I)—number of votes for candidate I, V—total number of votes, M (I)—candidate numbers for rank-ordering.

We have not tried here to make the programs particularly efficient or to make the I/O realistic. Rather, we have tried to show how program development proceeds, starting with an overall skeleton program and continuing through ever-increasing levels of detail. The language is a simple version of BASIC that should run on most computers.

Initial Program

Fig. 4 contains the initial program listing. The three major sections of the program—counting, ordering and output—have been replaced by program stubs that simply mark those sections that have been entered. We can test the overall program logic by entering a value for the number of candidates, C, and running the

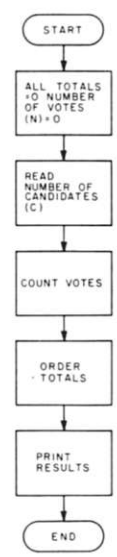


Fig. 3. Initial flowchart for the vote-analysis program.

```
LIST
10   DIM N(20),M(20)
15   REM NUMBER ØF  VØTES (V) = 0
20   LET V= 0
25   REM GET NUMBER ØF CANDIDATES (C)
30   PRINT "NUMBER ØF CANDIDATES = ";
35   INPUT C
40   REM CLEAR ALL VØTE CØUNTERS
45   FØR I= 1 TØ C+ 2
50   LET N(I)= 0
55   NEXT I
60   REM CØUNT VØTES
65   GØSUB 1000
70   REM ØRDER VØTE TØTALS
75   GØSUB 2000
80   REM ØUTPUT TØTALS
85   GØSUB 3000
999   END
1000   REM VØTE CØUNTING PRØGRAM
1010   PRINT "ATTEMPTED VØTE CØUNTING"
1020   RETURN
2000   REM TØTAL ØRDERING PRØGRAM
2010   PRINT "ATTEMPTED ØRDERING"
2020   RETURN
3000   REM ØUTPUT RØUTINE
3010   PRINT "REACHED ØUTPUT RØUTINE"
3020   RETURN
9999   END
RUN
NUMBER ØF CANDIDATES = ? 0
ATTEMPTED VØTE CØUNTING
ATTEMPTED ØRDERING
REACHED ØUTPUT RØUTINE
READY
```

Fig. 4. Initial listing for the vote-analysis program. All the subprograms are left as unexpanded stubs.

program (note the RUN results at the bottom). In fact, there was a slight error initially caused by the omission of the final END statement. This error was quickly corrected before any stubs were expanded.

The First Level of Expansion

Fig. 5 is the flowchart of the expanded vote-counting program. Here there are three cases to consider:

1. The last ballot (marked with the number C + 3) is not counted in the totals.
2. Blank ballots (marked by zero) are included in the total number of votes but are not credited to any category.
3. Other ballots must be credited to the appropriate category (i.e., to a candidate, write-in category or improperly marked category).

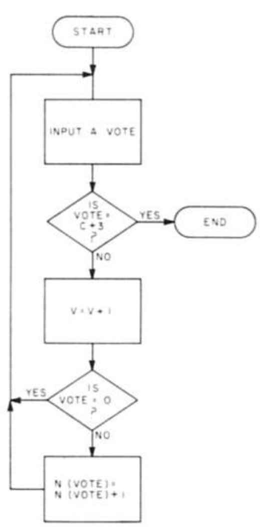


Fig. 5. Flowchart for the vote-counting subprogram.

Fig. 6 contains the BASIC program with the vote-counting stub expanded. We checked this program with the data in Example 1 (see the results at the bottom of Fig. 6).

Fig. 7 contains the BASIC program with the output stub expanded. This program was also checked with cases 1 and 2. Note the added statement

3020 IF C = 0 THEN 3045

This correction means that if there are no candidates, the program does not print headings, a list of candidates or vote totals. Note that the case without a candidate, although it seems useless, is by no means an uncommon situation in real elections, particularly at the local level. The results from this expanded program are in Fig. 8.

```
LIST
10   DIM N(20),M(20)
15   REM NUMBER ØF  VØTES (V) = 0
20   LET V= 0
25   REM GET NUMBER ØF CANDIDATES (C)
30   PRINT "NUMBER ØF CANDIDATES = ";
35   INPUT C
40   REM CLEAR ALL VØTE CØUNTERS
45   FØR I= 1 TØ C+ 2
50   LET N(I)= 0
55   NEXT I
60   REM CØUNT VØTES
65   GØSUB 1000
70   REM ØRDER VØTE TØTALS
75   GØSUB 2000
80   REM ØUTPUT TØTALS
85   GØSUB 3000
999   END
1000   REM VØTE CØUNTING PRØGRAM
1005   REM FETCH NEXT VØTE (J)
1010   PRINT "NEXT VØTE IS";
1015   INPUT J
1020   REM DØNE IF VØTE IS ENDING MARK (C+3)
1025   IF J=C+ 3 THEN 1065
1030   REM ADD VØTE TØ TØTAL (V)
1035   LET V=V+ 1
1040   REM IGNØRE VØTE IF BALLØT UNMARKD (J=0)
1045   IF J= 0 THEN 1010
1050   REM ADD VØTE TØ APPRØPRIATE TØTAL
1055   LET N(J)=N(J)+ 1
1060   GØTØ 1010
1065   RETURN
2000   REM TØTAL ØRDERING PRØGRAM
2010   PRINT "ATTEMPTED ØRDERING"
2020   RETURN
3000   REM ØUTPUT RØUTINE
3010   PRINT "REACHED ØUTPUT RØUTINE"
3020   RETURN
9999   END

RUN
NUMBER ØF CANDIDATES = ? 0
NEXT VØTE IS? 3
ATTEMPTED ØRDERING
REACHED ØUTPUT RØUTINE
READY

RUN
NUMBER ØF CANDIDATES = ? 1
NEXT VØTE IS? 1
NEXT VØTE IS? 4
ATTEMPTED ØRDERING
REACHED ØUTPUT RØUTINE
READY
```

Fig. 6. Listing for the vote-analysis program with the vote-counting subprogram expanded.

Fig. 9 is a flowchart for the first expansion of the rank-ordering routine. The idea is to keep interchanging pairs of elements until all pairs are in the correct order (i.e., largest number first). Flag F is cleared initially and set to 1 if an interchange is performed. So, if F = 1 at the end of a pass through the list, another pass is necessary. If F = 0 at the end, the list must be in order. Although this may appear an unsophisticated sorting method, it is perfectly acceptable for short lists like the ones handled by this program. The number of candidates in an election rarely exceeds ten. Note that no sorting is necessary if there is only one candidate or are none.

Fig. 10 is the BASIC program with the ordering routine ex-

```
LIST
10   DIM N(20),M(20)
15   REM NUMBER ØF VØTES (V) = 0
20   LET V= 0
25   REM GET NUMBER ØF CANDIDATES (C)
30   PRINT "NUMBER ØF CANDIDATES = ";
35   INPUT C
40   REM CLEAR ALL VØTE CØUNTERS
45   FØR I= 1 TØ C+ 2
50   LET N(I)= 0
55   NEXT I
60   REM CØUNT VØTES
65   GØSUB 1000
70   REM ØRDER VØTE TØTALS
75   GØSUB 2000
80   REM ØUTPUT TØTALS
85   GØSUB 3000
999  END
1000  REM VØTE CØUNTING PRØGRAM
1005  REM FETCH NEXT VØTE (J)
1010  PRINT "NEXT VØTE IS";
1015  INPUT J
1020  REM DØNE IF VØTE IS ENDING MARK (C+3)
1025  IF J=C+ 3 THEN 1065
1030  REM ADD VØTE TØ TØTAL (V)
1035  LET V=V+ 1
1040  REM IGNØRE VØTE IF BALLØT UNMARKED (J=0)
1045  IF J= 0 THEN 1010
1050  REM ADD VØTE TØ APPRØPRIATE TØTAL
1055  LET N(J)=N(J)+ 1
1060  GØTØ 1010
1065  RETURN
2000  REM TØTAL ØRDERING PRØGRAM
2010  PRINT "ATTEMPTED ØRDERING"
2020  RETURN
3000  REM ØUTPUT RØUTINE
3005  PRINT "NUMBER ØF CANDIDATES = ";C
3010  PRINT "NUMBER ØF VØTES = ";V
3015  REM SKIP CANDIDATE TØTALS IF NØ CANDIDATES
3020  IF C= 0 THEN 3045
3025  PRINT "CANDIDATE NUMBER    VØTE TØTAL"
3030  FØR I= 1 TØ C
3035  PRINT TAB( 5),I,TAB( 25),N(I)
3040  NEXT I
3045  PRINT "NUMBER ØF WRITE-INS = ";N(C+ 1)
3050  PRINT "NUMBER ØF IMPRØPER BALLØTS = ";N(C+ 2)
3055  RETURN
9999  END
```

Fig. 7. Listing for the vote-analysis program with the vote-counting and output subprograms expanded.

```
RUN
NUMBER ØF CANDIDATES = ? 0
NEXT VØTE IS? 3
ATTEMPTED ØRDERING
NUMBER ØF CANDIDATES =  0
NUMBER ØF VØTES =  0
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY

RUN
NUMBER ØF CANDIDATES = ? 1
NEXT VØTE IS? 1
NEXT VØTE IS? 4
ATTEMPTED ØRDERING
NUMBER ØF CANDIDATES =  1
NUMBER ØF VØTES =  1
CANDIDATE NUMBER    VØTE TØTAL
     1                  1
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY

RUN
NUMBER ØF CANDIDATES = ? 2
NEXT VØTE IS? 1
NEXT VØTE IS? 1
NEXT VØTE IS? 2
NEXT VØTE IS? 5
ATTEMPTED ØRDERING
NUMBER ØF CANDIDATES =  2
NUMBER ØF VØTES =  3
CANDIDATE NUMBER    VØTE TØTAL
     1                  2
     2                  1
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY
```

Fig. 8. Results from the program of Fig. 7.

CASE 1. NO CANDIDATES, NO VOTES
C = 0
V = 3 (ENDING MARKER)
CASE 2. ONE CANDIDATE, ONE VOTE
C = 0
V = 1
V = 4 (ENDING MARKER)

Example 1.

panded. Note that the interchange subroutine is left as a program stub. It will be expanded later. For some simple cases for checking this program, see Example 2. Fig. 11 shows the results from this program. Note that an interchange was attempted in Case 4, but not in Case 3.

The Second Level of Expansion

Fig. 12 shows the program with the interchange stub expanded. Statement 3035 now prints the identification number M(I), which is interchanged, but statements 2010 and 2033 had to be changed to give a value to M(I) when there is only one candidate.

Fig. 12 also contains a further expansion of the ordering routine (see flowchart in Fig. 13) to handle more efficiently the simple, but common, case where there are only two candidates. Further expansions could check for erroneous values of number of candidates

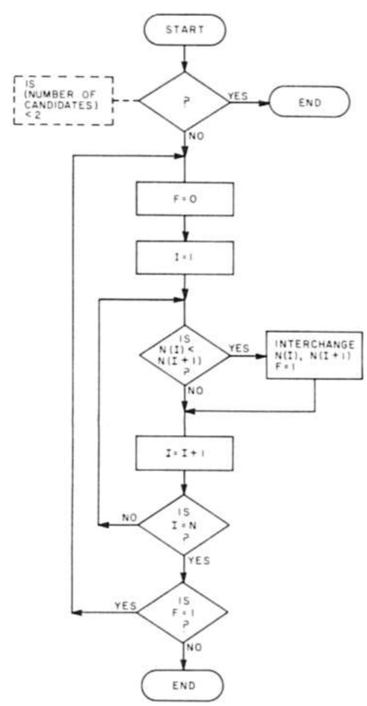


Fig. 9. Flowchart for the rank-ordering subprogram.

```
LIST
10   DIM N(20),M(20)
15   REM NUMBER ØF VØTES (V) = 0
20   LET V= 0
25   REM GET NUMBER ØF CANDIDATES (C)
30   PRINT "NUMBER ØF CANDIDATES = ";
35   INPUT C
40   REM CLEAR ALL VØTE CØUNTERS
45   FØR I= 1 TØ C+ 2
50   LET N(I)= 0
55   NEXT I
60   REM CØUNT VØTES
65   GØSUB 1000
70   REM ØRDER VØTE TØTALS
75   GØSUB 2000
80   REM ØUTPUT TØTALS
85   GØSUB 3000
999  END
1000  REM VØTE CØUNTING PRØGRAM
1005  REM FETCH NEXT VØTE (J)
1010  PRINT "NEXT VØTE IS";
1015  INPUT J
1020  REM DØNE IF VØTE IS ENDING MARK (C+3)
1025  IF J=C+ 3 THEN 1065
1030  REM ADD VØTE TØ TØTAL (V)
1035  LET V=V+ 1
1040  REM IGNØRE VØTE IF BALLØT UNMARKED (J=0)
1045  IF J= 0 THEN 1010
1050  REM ADD VØTE TØ APPRØPRIATE TØTAL
1055  LET N(J)=N(J)+ 1
1060  GØTØ 1010
1065  RETURN
2000  REM TØTAL ØRDERING PRØGRAM
2005  REM NØ ØRDERING NECESSARY IF ZERØ ØR ØNE CANDIDATES
2010  IF C< 2 THEN 2085
2015  REM ASSIGN MARKERS TØ CANDIDATES FØR SØRTING
2020  FØR I= 1 TØ C
2025  LET M(I)=I
2030  NEXT I
2035  REM SØRT VØTE TØTALS
2040  LET F= 0
2045  FØR I= 1 TØ C- 1
2050  REM CHECK IF TØTALS ARE IN ØRDER
2055  IF N(I)>=N(I+ 1) THEN 2070
2060  REM IF ØUT ØF ØRDER, INTERCHANGE PAIR
2065  GØSUB 2500
2070  NEXT I
2075  REM DØ ANØTHER PASS IF ANY INTERCHANGES ØCCURRED
2080  IF F= 1 THEN 2040
2085  RETURN
2500  REM INTERCHANGE TØTALS, MARKERS FØR ØRDERING
2510  PRINT "ATTEMPTED INTERCHANGE"
2520  RETURN
3000  REM ØUTPUT RØUTINE
3005  PRINT "NUMBER ØF CANDIDATES = ";C
3010  PRINT "NUMBER ØF VØTES = ";V
3015  REM SKIP CANDIDATE TØTALS IF NØ CANDIDATES
3020  IF C= 0 THEN 3045
3025  PRINT "CANDIDATE NUMBER    VØTE TØTAL"
3030  FØR I= 1 TØ C
3035  PRINT TAB( 5),I,TAB( 25),N(I)
3040  NEXT I
3045  PRINT "NUMBER ØF WRITE-INS = ";N(C+ 1)
3050  PRINT "NUMBER ØF IMPRØPER BALLØTS = ";N(C+ 2)
3055  RETURN
9999  END
```

Fig. 10. Listing of vote-analysis program with all subprograms expanded by one level.

```
RUN
NUMBER ØF CANDIDATES = ? 0
NEXT VØTE IS? 3
NUMBER ØF CANDIDATES =  0
NUMBER ØF VØTES =  0
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY


RUN
NUMBER ØF CANDIDATES = ? 1
NEXT VØTE IS? 1
NEXT VØTE IS? 4
NUMBER ØF CANDIDATES =  1
NUMBER ØF VØTES =  1
CANDIDATE NUMBER    VØTE TØTAL
     1                   1
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY


RUN
NUMBER ØF CANDIDATES = ? 2
NEXT VØTE IS? 1
NEXT VØTE IS? 2
NEXT VØTE IS? 2
NEXT VØTE IS? 5
ATTEMPTED INTERCHANGE
NUMBER ØF CANDIDATES =  2
NUMBER ØF VØTES =  3
CANDIDATE NUMBER    VØTE TØTAL
     1                   1
     2                   2
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY
```

Fig. 11. Results from the program of Fig. 10.

CASE 3. TWO CANDIDATES. THREE VOTES (2 FOR NUMBER 1, 1 FOR NUMBER 2)
C = 2
V = 1
V = 1
V = 2
V = 5 (ENDING MARKER)
CASE 4. TWO CANDIDATES, THREE VOTES (1 FOR NUMBER 1, 2 FOR NUMBER 2)
C = 2
V = 1
V = 2
V = 2
V = 5 (ENDING MARKER)

Example 2.

(less than zero or more than the program can handle) and erroneous data (values that are undefined). Other expansions could check for ties, handle cases where more than one vote is allowed (e.g., vote for four of the above) and identify the ballots on which write-ins were marked.

Conclusion

Top-down design is a method for designing, debugging and testing large programs. It requires the programmer to start with the overall program logic and to continue expanding subprograms until the task is fully defined. Each level is checked in its actual working environment before the next level is attempted. Thus, integration of modules and system-level debugging and testing are performed throughout program development rather than all at the end. Program stubs replace unexpanded programs or modules at each level. Top-down design is a systematic approach to writing large programs. Personal computer users should carefully consider its use when attempting complex projects.■

References

1. J. K. Hughes and J. J. Michtom, *A Structured Ap-*

```
LIST
10   DIM N(20),M(20)
15   REM NUMBER ØF VØTES (V) = 0
20   LET V= 0
25   REM GET NUMBER ØF CANDIDATES (C)
30   PRINT "NUMBER ØF CANDIDATES = ";
35   INPUT C
40   REM CLEAR ALL VØTE CØUNTERS
45   FØR I= 1 TØ C+ 2
50   LET N(I)= 0
55   NEXT I
60   REM CØUNT VØTES
65   GØSUB 1000
70   REM ØRDER VØTE TØTALS
75   GØSUB 2000
80   REM ØUTPUT TØTALS
85   GØSUB 3000
999  END
1000 REM VØTE CØUNTING PRØGRAM
1005 REM FETCH NEXT VØTE (J)
1010 PRINT "NEXT VØTE IS";
1015 INPUT J
1020 REM DØNE IF VØTE IS ENDING MARK (C+3)
1025 IF J=C+ 3 THEN 1065
1030 REM ADD VØTE TØ TØTAL (V)
1035 LET V=V+ 1
1040 REM IGNØRE VØTE IF BALLØT UNMARKED (J=0)
1045 IF J= 0 THEN 1010
1050 REM ADD VØTE TØ APPRØPRIATE TØTAL
1055 LET N(J)=N(J)+ 1
1060 GØTØ 1010
1065 RETURN
2000 REM TØTAL ØRDERING PRØGRAM
2005 REM DØNE IF NØ CANDIDATES
2010 IF C= 0 THEN 2085
2015 REM ASSIGN MARKERS TØ CANDIDATES FØR SØRTING
2020 FØR I= 1 TØ C
2025 LET M(I)=I
2030 NEXT I
2031 REM NØ ØRDERING NECESSARY IF ØNLY ØNE CANDIDATE
2033 IF C= 1 THEN 2085
2035 REM SØRT VØTE TØTALS
2036 REM HANDLE CASE ØF ØNLY TWØ CANDIDATES SEPARATELY
2038 IF C= 2 THEN 2090
2040 LET F= 0
2045 FØR I= 1 TØ C- 1
2050 REM CHECK IF TØTALS ARE IN ØRDER
2055 IF N(I)>=N(I+ 1) THEN 2070
2060 REM IF ØUT ØF ØRDER, INTERCHANGE PAIR
2065 GØSUB 2500
2070 NEXT I
2075 REM DØ ANØTHER PASS IF ANY INTERCHANGES ØCCURRED
2080 IF F= 1 THEN 2040
2085 RETURN
2090 REM ØRDER TØTALS FØR TWØ CANDIDATES ØNLY
2095 REM NØ PRØBLEM IF ALREADY IN ØRDER
2100 IF N( 1)>=N( 2) THEN 2120
2105 REM IF ØUT ØF ØRDER, INTERCHANGE
2110 LET I= 1
2115 GØSUB 2500
2120 RETURN
2500 REM INTERCHANGE TØTALS, MARKERS FØR ØRDERING
2505 REM MARK THAT INTERCHANGE ØCCURRED (F=1)
2510 LET F= 1
2515 REM INTERCHANGE TØTALS
2520 LET T=N(I)
2525 LET N(I)=N(I+ 1)
2530 LET N(I+ 1)=T
2535 REM INTERCHANGE MARKERS
2540 LET T=M(I)
2545 LET M(I)=M(I+ 1)
2550 LET M(I+ 1)=T
2555 RETURN
3000 REM ØUTPUT RØUTINE
3005 PRINT "NUMBER ØF CANDIDATES = ";C
3010 PRINT "NUMBER ØF VØTES = ";V
3015 REM SKIP CANDIDATE TØTALS IF NØ CANDIDATES
3020 IF C= 0 THEN 3045
3025 PRINT "CANDIDATE NUMBER     VØTE TØTAL"
3030 FØR I= 1 TØ C
3035 PRINT TAB( 5),M(I),TAB( 25),N(I)
3040 NEXT I
3045 PRINT "NUMBER ØF WRITE-INS = ";N(C+ 1)
3050 PRINT "NUMBER ØF IMPRØPER BALLØTS = ";N(C+ 2)
3055 RETURN
9999 END
```

Fig. 12. Listing of vote-analysis program with improved rank-ordering subprogram. The subprogram now handles the case of two candidates more efficiently.

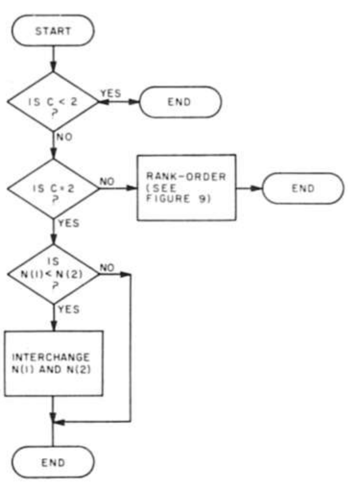


Fig. 13. Flowchart of the improved rank-ordering subprogram.

```
RUN
NUMBER ØF CANDIDATES = ? 2
NEXT VØTE IS? 1
NEXT VØTE IS? 1
NEXT VØTE IS? 2
NEXT VØTE IS? 5
NUMBER ØF CANDIDATES =  2
NUMBER ØF VØTES =  3
CANDIDATE NUMBER     VØTE TØTAL
    1                    2
    2                    1
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY


RUN
NUMBER ØF CANDIDATES = ? 2
NEXT VØTE IS? 1
NEXT VØTE IS? 2
NEXT VØTE IS? 2
NEXT VØTE IS? 5
NUMBER ØF CANDIDATES =  2
NUMBER ØF VØTES =  3
CANDIDATE NUMBER     VØTE TØTAL
    2                    2
    1                    1
NUMBER ØF WRITE-INS =  0
NUMBER ØF IMPRØPER BALLØTS =  0
READY
```

Fig. 14. Results from the program of Fig. 12.

proach to Programming, Prentice-Hall, Englewood Cliffs NJ, 1977.

2. E. Yourdon, *Techniques of Program Structure and Design,* Prentice-Hall, Englewood Cliffs NJ, 1975.

3. B. W. Kernighan and P. J. Plauger, *The Elements of Programming Style,* McGraw-Hill, NY, 1974.

4. E. J. Miller, Jr., and G. E. Lindamood, "Structured Programming: Top-down Approach," *Datamation,* December 1973, pp. 55-57.

5. R. W. Ulrickson, "Solve Software Problems Step by Step," *Electronic Design,* January 18, 1977, pp. 54-58.

6. L. A. Leventhal, *8080A/8085 Assembly Language Programming,* Osborne and Associates, Berkeley CA, 1978.

The North Star Floppy System

an 11-year-old can build it!

Howie DiBlasi
Director, Vocational Education
Lake Havasu High School
Lake Havasu AZ 86403

Hi. My name is Mark; I am 11 years old. I just finished a North Star Floppy Disk Kit. It was easy; I really made it. And guess what? It worked the first time I hooked it up!"

Mark looked at me and smiled. He was really proud of himself, and I was too. If an 11-year-old can put the North Star Kit together, so can you.

"Hey, Dad, am I going to be rich and famous because I put the North Star together and you are writing about me in *Kilobaud*?"

I laughed. Rich? No. Famous? No. Proud and satisfied? Yes.

Here We Go

I ordered the North Star Kit and received it in a week from the Byte Shop in Phoenix. When I opened the box and examined the contents, I was impressed with the quality of the circuit boards and parts. All the parts were there, and complete instructions were included.

After looking over the instruction manual, I had my son read it to see if he understood what to do. He said, "No sweat," and at that point I decided to let him go ahead and build the kit.

Printed Matter

Four instruction manuals came with the kit: (1) Minifloppy Diskette Storage Drive OEM Manual; (2) North Star Disk Operating System Manual; (3) North Star BASIC Manual; (4) North Star MICRO-DISK SYSTEM MDS-A Instruction Manual.

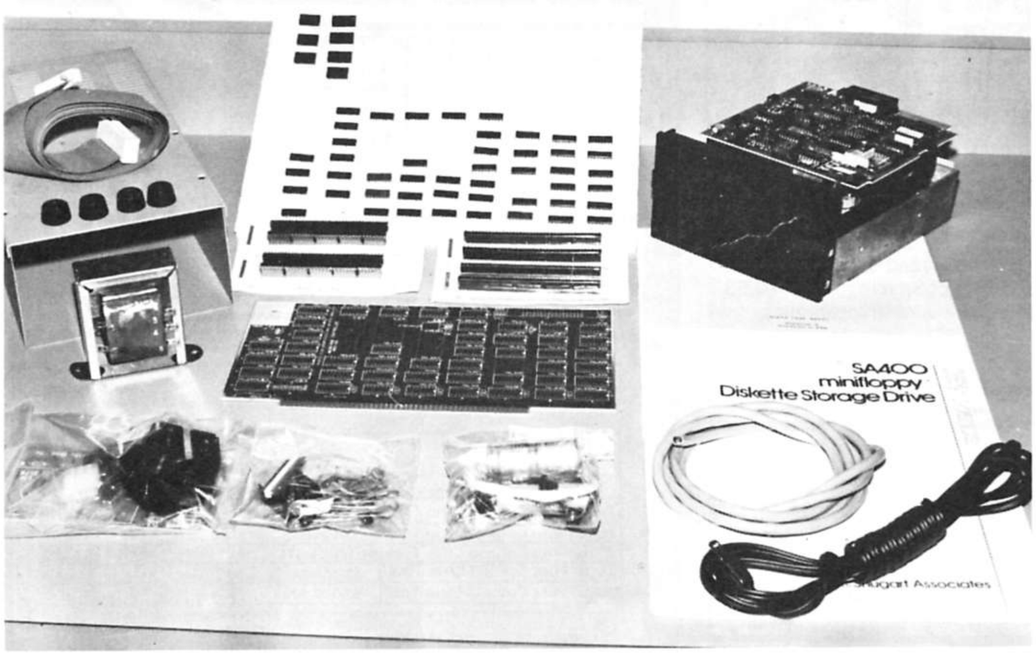
The instruction manual is divided into three sections: theory of operation, assembly instructions and system integration and schematics. The manuals are all well written and detail numerous situations and how to set things up. It was a pleasure to read through and understand the material. Right on, North Star!

Assembly

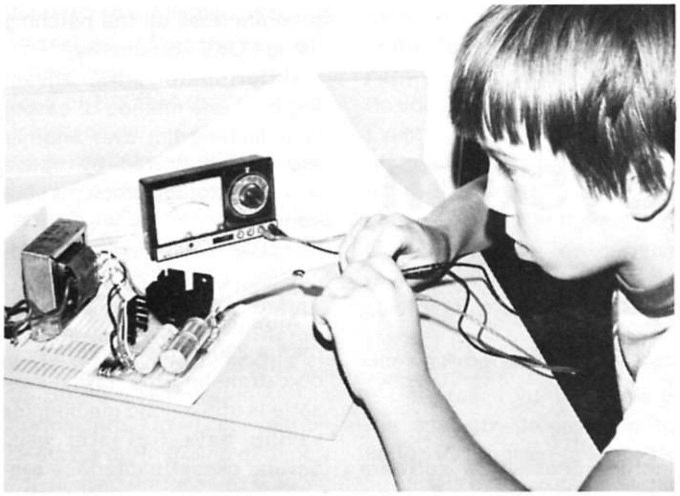
All parts were checked off by Mark, which helped him become familiar with the parts and learn their use. As he checked them, I took a few moments to explain the function of the various parts. Everything was there. Some kits don't always include all items; but North Star has it all together.

Mark installed the 47 IC sockets and soldered them in place. He had soldered a few times before so he was familiar with the correct circuit-board soldering procedure. He had a few problems with bridges, but a little Solder-Wick removed them. I was pleased to see a very professional soldermasked board; properly soldermasking a board helps to eliminate problems.

The eight resistors and 40



What you see is what you get. The kit comes complete for the North Star Disk System. The Shugart disk drive (back right) comes complete and assembled.



WOW! Five volts. After the power supply was completed, the connector plug was checked for correct voltages. All OK.

Look Dad, I did it! A very proud young man. If he can build the North Star System, you can. Let's go.

capacitors were then soldered to the board, and the crystal, 5 volt regulator and heat-sink hardware followed. It was now necessary to solder a 34 pin cable connector to the board. The MDS Controller board was plugged into the computer. Holding his breath, Mark connected the meter, which read + 5 volts. So far so good.

IC Installation

Mark watched while I demonstrated the correct way to install the ICs in the sockets. I made a quick check to make sure he had them in the correct location. The manual then gave two detailed pages of instruction for waveforms on a scope. Since I did not have a scope available we skipped this step.

Power Board Assembly

The disk drive can receive power three ways: (1) From +5 and +12 volts from an existing power supply; (2) power PC board to regulate power from an existing unregulated power supply; (3) North Star power-supply option (MDS-PS).

Since I knew we would be using the North Star with two different computers from one time to another, I had purchased the North Star power-supply option. Mark mounted the transformer in the cabinet and hooked up the wires, switch and fuse to complete the power supply. Ready to test the power supply for +5 and +12 volts at the power plug, Mark hooked up the meter and checked for the proper voltages. To our satisfaction, they were OK.

The last thing to do was to make two trace cuts on the MDS controller board and install two jumper wires. Done!

Final Check

The real test was drawing near. Mark installed the MDS controller board in the computer and hooked up the cables. With the power switch and computer on and the disk in the disk drive, Mark typed EX E900 and hit return. As he did that I explained that an asterisk on the screen signaled that everything was OK. The next command was GO BASIC. Mark did that and BASIC was loaded in 2 seconds. READY appeared on the screen and we were ready to program.

Up And Running

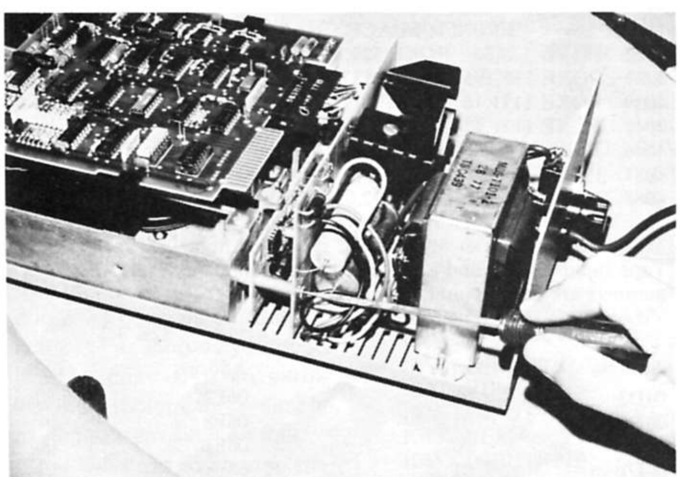
Mark and I input a small program to make sure everything was OK. It was. We sat at the computer for over three hours inputting programs and running them. It was getting late, so we stopped and decided to input some more programs during the next few days.

Summary

Total construction time for the project was 4 hours and 20 minutes. You could probably complete it in less time if you have experience building kits. Mark took his time building the kit, but the time spent paid off because the system worked the first time.

While Mark was running a few programs, I looked over the manuals. North Star BASIC is an extended version and has numerous functions. It also has an edit function to correct errors; it is a joy to use.

The OEM manual gives complete and detailed description of the disk drive and complete schematics. The North Star Disk Operating Systems Manual features complete instructions and operations for the DOS. It contains descriptions on creating files, types of files, deleting files, jump routines, read and write and many more procedures that are available for use. All the manuals are written so you can understand them. Maybe some other manufacturers will take a lesson from North Star. ■



Disk drive assembly. The power supply is assembled to the disk drive assembly with two spacers and screws on each side. The unit is then connected to the case.

A Simple Mailing System

a money-making time-saver

Stephen Gibson
PO Box 38386
Los Angeles CA 90038

One of the first tasks a small businessman wants his new computer to do is handle the company mailing list. A review of the many programs available reveals a big problem: Unless you have a disk storage system, you are forever condemned to load all those names and addresses via the DATA statement.

To read out the list or print labels, the data is usually read into a set of variables, then formatted to fit your particular hard-copy printer. If you fill your memory or want separate lists, you have to write a whole new program. To update, you are forced to list the program to find where the last DATA statement ended, then change the read routine.

All this nonsense takes valuable time and makes you a slave to the machine. It would be easier to write the program only once, and simply change the lists. Here are two ways to do it: (1) the cassette method and (2) the bare-bones method. At least one is bound to work for you.

Sneaky Software Secrets Revealed

The problem is not how to structure the ideal list program in BASIC, but how to save the names and addresses in a language that doesn't know how to save variables. The main program should have to be saved only once.

Surprisingly enough, a number of rather clever techniques have been developed to solve this problem. One method breaks down the name, a string variable, and feeds it to tape as a series of OUT statements. Another method uses the tape interface hard-wired in parallel with the terminal I/O. Still another chooses a software method by patching BASIC's terminal I/O over to the tape interface port, and then outputs the list via PRINT statements as though it were the terminal . . . very clever, because the program does all the patching using POKE statements.

Unfortunately, the advantages of one method of saving your mailing list over another are overshadowed by speed and tape storage problems with your unit.

Is Speed Your Thing?

Although somewhat slower than a 250K bit/second floppy-disk transfer, the lowly cassette is still a good medium for saving data for later use. Several cassette interface systems are available. They differ widely with respect to speed.

I picked the Tarbell high-speed interface and coupled it to the Data Duffer (see *Kilobaud*, March 1978, "Hear It and

Add these lines to the program to let the computer tell you when to abbreviate.

```
12   A = 20 :REM WIDTH OF TTY LABEL
1050  IF LEN(NA$(N)) > A THEN GOSUB 5000 : GOTO 1040
1060  IF LEN(CO$(N)) > A THEN GOSUB 5000 : GOTO 1055
1070  IF LEN(AD$(N)) > A THEN GOSUB 5000 : GOTO 1065
1080  IF LEN(CS$(N)) > A THEN GOSUB 5000 : GOTO 1075
1090  IF LEN(ZP$(N)) > A THEN GOSUB 5000 : GOTO 1085
5000  REM
5005  REM LINE LENGTH ERROR
5010  REM
5015  PRINT:PRINT"LINE TOO LONG!!":PRINT:RETURN
```

Add these lines to run the program with Mits 3.2 12K BASIC

```
3020  POKE 1776,110 : POKE 1778,32 : POKE 1784,1
3300  POKE 1776,0 : POKE 1778,128 : POKE 1784,1
4015  POKE 1787,110 : POKE 1789,16 : POKE 1794,111
4020  POKE 1778,0 : POKE 1784,255
4090  POKE 1787,0 : POKE 1789,1 : POKE 1794,1
4095  POKE 1778,128 : POKE 1784,1
```

Add these lines to modify Mits 3.2 8K BASIC to recognize leading spaces.

```
13   SP$ = " " :REM A SPACE CHARACTER
4022  POKE 528,54 : POKE 529,32 : POKE 530,35
4023  POKE 531,195 : POKE 532,224 : POKE 533,7
4039  POKE 1171,16 : POKE 1172,2
4041  POKE 1171,224 : POKE 1172,7
4046  IF B$ = (E$ + SP$) THEN 4090
4087  POKE 528,0 : POKE 529,0 : POKE 530,0
4088  POKE 531,0 : POKE 532,0 : POKE 533,0
```

Make these patches to Mits BASIC if you get hung up in the Tape Input routine and need to return to command level. All numbers are hexadecimal.

8K 3.2		12K 3.2	
Address	Byte	Address	Byte
04D3	80	06F2	80
04D9	01	06F8	01
04DC	00	06FB	00
04DE	01	06FD	01
04E3	01	0702	01

Fig. 1. Mits BASIC patches.

The entire system here is an Imsai 8080 with 24K of memory, ADM-3 terminal, Data Duffer, Teletype . . . and one efficient secretary.

See It!") as a reliable way to use cassettes without the hassle of a seemingly endless wait for a load or the fear that data was lost because a switch was off or a knob twisted the wrong way. The Tarbell manual suggests a variable-saving method in which the terminal I/O is software patched to the cassette I/O for a transfer. The routines in the mailing-list program make these patches to Mits 3.2 8K BASIC (see Program A). The normal Mits TTY I/O convention of status port "0" and data port "1" is used. Patches to Mits 3.2 12K BASIC are also listed in Fig. 1. If you don't have Mits BASIC or a Tarbell, there's still hope; you can use the bare-bones method described later.

Hard-Copy Hassles

Registration is the key ingredient for alignment of the labels on your printer. A sprocketed feed mechanism is almost a necessity. Of course, you can simply cut your labels out with a large paper-cutter, but the peel-off-type labels are more convenient and better looking. You need the sprocket feed to make them work properly. You might even consider custom labels with fancy artwork or the company logo.

I had quite a time finding off-the-shelf labels for my old sprocket-fed Teletype. Almost everyone sells ready-made forms for larger printers. There are a few companies that specialize in stock or custom labels from camera-ready artwork (see accompanying "Sources for More Information").

If you do start with a Teletype, by all means change the ribbon! Use a carbon ribbon rather than the stock cloth one —the printing looks so much better. Unique type fonts are also available for the Teletype. Even the Teletype can be made to look as good as an IBM Selectric . . . as long as you don't mind all caps—not an earth-shaking problem for a simple mailing system such as this.

You will have to change the platen if your Teletype is a friction-feed model. The modification to your machine is simple and inexpensive. I'm not advocating the Teletype as the ideal printer for this system; my company just happens to have one. Besides being slow, it's noisy! Eventually, I had to stick ours off in a room by itself to drown out the clatter. The advantages, of course, are that the machine is reliable and inexpensive. Used machines abound, and service is readily available.

Simple Program Does It All

Only four routines make up the cassette program. In the listing in Program A, lines 1 to 50 initialize the program. A generous 10,000 bytes are cleared away based on an average line length of 20 characters, with 5 lines given to each company and a list size of 100 companies. The variables S and L represent the maximum size of the list and the current list size, respectively. The subscripted variables in line 25 are dimensioned to the size of the list. Of course, you can set this value higher for a larger list if your memory capacity will permit it. The command level routine prints suitable prompts for those unfamiliar with the program. A branch is made at line 155 based on the value of C.

To enter names at line 1000, the list counter L is incremented by 1 and a test is made to see if the list size is greater than 100 names. It might be later on, so we must check it out. If so, the list counter is decremented back to 100 and a return is made to the command routine. In line 1030, a message indicates that the number symbol (#) can be used to exit the routine. A FOR/NEXT loop inputs the names and addresses into the subscripted variables.

You might wish to make the prompts different for your version. Instead of "ZIP . . . ," for instance, you might want the program to print "COUNTRY . . . ," if you mail overseas. Or you could eliminate "ZIP" (ZP$) altogether and squeeze it into the CITY/STATE line.

If # is typed in line 1045, a branch is made and the list

```
1 REM ----------------------------------------------------------------
2 REM              **** MAILING LIST ****
3 REM ----------------------------------------------------------------
4 REM  BY STEPHEN GIBSON 1/10/77
5 REM  RUNS ONLY ON MITS 3.2 8K BASIC
6 REM  AND TARBELL CASSETTE INTERFACE
7 REM ----------------------------------------------------------------
8 REM  INITIALIZE
9 REM ----------------------------------------------------------------
10 CLEAR 10000 :REM CLEAR SPACE FOR LIST
15 S = 100 :REM MAXIMUM LIST SIZE
20 L = 0 :REM CURRENT LIST SIZE
25 DIM NA$(S),CO$(S),AD$(S),CS$(S),ZP$(S)
30 E$ = "#"  :REM END OF LIST CHARACTER
35 OUT 1,26  :REM CLEARS SCREEN
40 PRINTTAB(20);"*** THIS IS MAILING LIST ***"
50 PRINT
100 REM ---------------------------------------------------------------
105 REM  COMMAND LEVEL ROUTINE
110 REM ---------------------------------------------------------------
115 PRINT"PLEASE ENTER YOUR COMMAND:":PRINT
120 PRINT"ENTER NAMES INTO LIST    = 1"
125 PRINT"PRINT-OUT OF LIST        = 2"
130 PRINT"STORE LIST ON TAPE       = 3"
135 PRINT"READ LIST FROM TAPE      = 4"
140 PRINT
145 INPUT"COMMAND";C
150 IF C > 4 THEN 115
155 ON INT(C) GOTO 1000, 2000, 3000, 4000
1000 REM --------------------------------------------------------------
1005 REM  ENTER NAMES ROUTINE
1010 REM --------------------------------------------------------------
1015 L=L+1
1020 IF L > 100 THEN 1400
1025 PRINT"IF YOU WISH TO EXIT THIS ROUTINE . . . "
1030 PRINT"TYPE ONE OF THESE '#',THEN 'RETURN'."
1035 FOR N = L TO 100 :PRINT:PRINT"NUMBER ";N:PRINT
1040 INPUT"NAME :             ";NA$(N)
1045 IF NA$(N) = "#" THEN 1300
1047 IF NA$(N) = "\" THEN N=N-2 :GOTO 1100
1050 REM
1055 INPUT"COMPANY :          ";CO$(N)
1060 REM
1065 INPUT"ADDRESS :          ";AD$(N)
1070 REM
1075 INPUT"CITY & STATE :  ";CS$(N)
1080 REM
1085 INPUT"ZIP :              ";ZP$(N)
1100 NEXT
1200 L = 100 : GOTO 1500
1300 L = N - 1 : GOTO 1600
1400 L = L - 1
1500 PRINT:PRINT"THE LIST IS FULL.":PRINT
```

```
1600 PRINT:PRINT"YOU HAVE ";L;"NAMES ON THIS LIST."
1700 GOTO 100
2000 REM ----------------------------------------------------------------
2005 REM  PRINT-OUT ROUTINE
2010 REM ----------------------------------------------------------------
2015 REM  PRINT
2020 PRINT"1) LINE UP LABELS IN PRINTER."
2025 PRINT
2030 PRINT"2) TURN ON PRINTER."
2035 PRINT
2040 PRINT"3) TYPE ANY LETTER, THEN 'RETURN'."
2045 PRINT:INPUT"WAITING . . . ";W$
2050 FOR X = 1 TO L STEP 3
2055 Y = X + 1 : Z = X + 2
2060 PRINT TAB(0) ; NA$(X) ; TAB(25) ; NA$(Y) ; TAB(51) ; NA$(Z)
2065 PRINT TAB(0) ; CO$(X) ; TAB(25) ; CO$(Y) ; TAB(51) ; CO$(Z)
2070 PRINT TAB(0) ; AD$(X) ; TAB(25) ; AD$(Y) ; TAB(51) ; AD$(Z)
2075 PRINT TAB(0) ;  CS$(X) ; TAB(25) ;  CS$(Y) ; TAB(51) ;  CS$(Z)
2080 PRINT TAB(0) ;  ZP$(X) ; TAB(25) ;  ZP$(Y) ; TAB(51) ;  ZP$(Z)
2085 PRINT:PRINT
2090 NEXT
2095 GOTO 100
3000 REM ----------------------------------------------------------------
3005 REM  STORE ON TAPE ROUTINE
3010 REM ----------------------------------------------------------------
3011 PRINT:PRINT"1) PLACE NEW CASSETTE IN RECORDER."
3012 PRINT:PRINT"2) PUT IN RECORD MODE AND ZERO COUNTER."
3013 PRINT:PRINT"3) WAIT A FEW SECONDS TO ALLOW A LEADER."
3014 PRINT:INPUT"4) TYPE ANY LETTER, THEN 'RETURN'.";W$
3015 S$ = CHR$(195) + CHR$(230)
3020 POKE 1233,110 : POKE 1235,32 : POKE 1241,111
3025 FOR N = 1 TO L
3030 D$(1) = NA$(N)
3035 D$(2) = CO$(N)
3040 D$(3) = AD$(N)
3045 D$(4) = CS$(N)
3050 D$(5) = ZP$(N)
3055 FOR J = 1 TO 5
3060 FOR K = 1 TO 100 : NEXT K
3065 B$ = S$ + D$(J)
3070 PRINT B$
3075 NEXT J
3080 NEXT N
3085 FOR T = 1 TO 3
3090 B$ = S$ + E$
3095 FOR K = 1 TO 100 : NEXT K
3100 PRINT B$
3200 NEXT T
3300 POKE 1233,0 : POKE 1235,128 : POKE 1241,1
3400 GOTO 100
4000 REM ----------------------------------------------------------------
4005 REM  READ FROM TAPE ROUTINE
4010 REM ----------------------------------------------------------------
4011 PRINT:PRINT"1) PLACE CASSETTE IN RECORDER."
4012 PRINT:PRINT"2) SET COUNTER AND PUSH PLAY."
4013 PRINT:PRINT"3) ALLOW TIME FOR LEADER."
4014 PRINT:INPUT"4) TYPE ANY LETTER, THEN 'RETURN'.";W$
4015 POKE 1244,110 : POKE 1246,16 : POKE 1251,111
4020 POKE 1235,0 : POKE 1241,255
4025 FOR N = 1 TO 101
4030 FOR J = 1 TO 5
4035 OUT 110,16
4040 INPUT B$
4045 IF B$ = E$ THEN 4090
4050 D$(J) = B$
4055 NEXT J
4060 NA$(N) = D$(1)
4065 CO$(N) = D$(2)
4070 AD$(N) = D$(3)
4075 CS$(N) = D$(4)
4080 ZP$(N) = D$(5)
4085 NEXT N
4090 POKE 1244,0 : POKE 1246,1 : POKE 1251,1
4095 POKE 1235,128 : POKE 1241,1
4100 L = N - 1
4200 PRINT:PRINT"THIS LIST HAS ";L;" NAMES ON IT.":PRINT
4300 GOTO 100
```

Program A. Program listing for A Simple Mailing System. Here are the routines you need to patch Mits 8K 3.2 BASIC to load or save your mailing list using the Tarbell high-speed cassette interface.

counter L is decremented by one (1) and a return is made to the command routine. Sometimes I make mistakes when entering a name (my secretary never does). I find it convenient to be able to type a character that tells the program to go back one name and start over. Line 1047 does it all. I chose a backslash, but you should feel free to choose your own character to personalize this program. You could insert this line after every input if you'd rather check your work a line at a time.

Another useful addition is to print a space, for example, where the name goes in the event you have a company name, but no one individual to mail to. A space is a logical entry. Don't try it unless you add the appropriate lines from Fig. 1 because Mits BASIC ignores leading spaces. The listed POKEs change the input routine to add a space if a carriage return is received. I found it convenient to print the current list size in line 1600 before exiting this routine.

The printout routine must be tailored to your particular printer. The program format given is for a standard Teletype using peel-off labels spaced three across. Lines 2020 to 2045 give instructions. The variable W$ is only a buffer to wait until you are ready to print. Extra PRINT statements in line 2085 advance the form to the next set of labels. To print your labels three at a time for the popular machine-gun mailings, simply substitute the lines in Fig. 2.

The store (on tape) routine at line 3000 begins the really useful aspects of this program. It is here that the names and addresses only are fed to tape. Instructions are given in lines 3011 to 3014. W$ is still only a wait buffer. S$ is set to the value of the Tarbell start and sync bytes. POKEs to Mits BASIC are then made in line 3020 to shift the terminal I/O to the cassette I/O port. The names and addresses are placed in a D$ buffer, then output with the start and sync bytes as B$ via PRINT statements.

Instead of this format . . .

John Craig	Wayne Green	Stephen Gibson
Editor	Publisher	Famous author
Kilobaud Magazine	Kilobaud Magazine	PO Box 38386
Peterborough NH	Peterborough NH	Los Angeles CA
03458	03458	90038

You might want this . . .

John Craig	John Craig	John Craig
Editor	Editor	Editor
Kilobaud Magazine	Kilobaud Magazine	Kilobaud Magazine
Peterborough NH	Peterborough NH	Peterborough NH
03458	03458	03458
Wayne Green	Wayne Green	Wayne Green
Publisher	Publisher	Publisher
Kilobaud Magazine	Kilobaud Magazine	Kilobaud Magazine
Peterborough NH	Peterborough NH	Peterborough NH
03458	03458	03458

Then substitute these lines . . .

```
2050  REM  3-UP FORMAT
2055  FOR N = 1 TO L
2060  PRINT TAB(0);NA$(N);TAB(25);NA$(N);TAB(51);NA$(N)
2065  PRINT TAB(0);CO$(N);TAB(25);CO$(N);TAB(51);CO$(N)
2070  PRINT TAB(0);AD$(N);TAB(25);AD$(N);TAB(51);AD$(N)
2075  PRINT TAB(0); CS$(N);TAB(25); CS$(N);TAB(51); CS$(N)
2080  PRINT TAB(0); ZP$(N);TAB(25); ZP$(N);TAB(51); ZP$(N)
```

Fig. 2. Instead of this format . . .

The delay loop in line 3060 bears some explanation. When data is brought back into the program, allow time for BASIC to reinsert the data into the appropriate subscripted variables by implementing a delay during the output sequence. You could, perhaps, shorten the delay, but you might lose some of your data. A value of 100 for T allows plenty of safety.

The End of List character, E$, must also be output. The computer will look for this character when the list is played back into the machine to set the list counter. This particular arrangement allows lists of varying size and the addition of more names to a short list.

Beginning on line 3090, E$ is linked to the start and sync bytes and output three times. Why three; isn't once enough? That's right. But suppose you had a dropout on the tape. It does happen on old cassettes, particularly cheap ones. Even if you use top-notch cassettes, you may still lose a byte because your recorder's slow AGC attack time may turn the beginning of a byte to garbage. I proved it writing this program.

The computer missed the E$ on playback. It just sat there waiting. It was very annoying . . . especially because the program had POKEd the I/O away from my terminal to the cassette interface. I had no way to talk to my machine except via the system monitor and the front panel to patch things up between my computer and its program. The pandemic Murphy's Law says you won't need to use the patches I made if I list them in Fig. 1. I output the E$ three times, rather than once, and beat old Edsel Murphy by even a New York second! (That's easy for me to say, you say.) The routine ends by POKEing BASIC back to normal I/O and jumping to the command routine.

The tape input routine is very similar. Instructions are given in lines 4011 to 4014. The I/O is POKEd to the cassette port just as before, and data is input by another FOR/NEXT loop. It is useful to print out the size of the list after the I/O is POKEd back because not all lists will be set at the maximum size. You will then be able to add to the current list by using the input routine. Then save the whole thing as a full list.

The Bare-Bones Method

Suppose you have a computer and a Teletype, but neither speaks Mits BASIC nor recognizes Tarbell format. If your Teletype has a paper-tape punch (most do), you can still benefit from this system.

Start by making those nifty mods to the Teletype, especially the ribbon. Then enter the program in Program B. The variables are the same as the cassette program, but the prompts are different and the save and read routines are left out.

Next, run the program and enter the names and addresses. When you print the list, simply turn on the paper-tape punch at the same time. You will have an exact copy of the printout, as well as a set of labels, on paper tape. You can then reprint the list by using the Teletype in the local mode and reading off the paper tape. Turn on the punch again while printing if you need a spare copy of your list. Use a separate punch if you have one.

```
1   REM  -----------------------------------------------------------------
2   REM        **** BARE BONES MAILING LIST ****
3   REM  -----------------------------------------------------------------
4   REM  BY STEPHEN GIBSON 12/11/76
5   REM  RUNS ON ASR-33 TTY OR SIMILAR
6   REM  PRINTER WITH PAPER TAPE PUNCH
7   REM  -----------------------------------------------------------------
8   REM  INITIALIZE
9   REM  -----------------------------------------------------------------
10  CLEAR 10000 :REM CLEAR SPACE FOR LIST
15  S= 100 :REM MAXIMUM LIST SIZE
20  L = 0 :REM CURRENT LIST SIZE
25  DIM NA$(S),CO$(S),AD$(S),CS$(S),ZP$(S)
30  E$ = "#" :REM END OF LIST CHARACTER
35  OUT 1,26 :REM CLEARS SCREEN
40  PRINTTAB(20);"*** THIS IS MAILING LIST ***"
50  PRINT
100  REM  ---------------------------------------------------------------
105  REM  COMMAND LEVEL ROUTINE
110  REM  ---------------------------------------------------------------
115  PRINT"PLEASE ENTER YOUR COMMAND:":PRINT
120  PRINT"ENTER NAMES INTO LIST   = 1"
125  PRINT"PRINT-OUT OF LIST       = 2"
140  PRINT
```

```
145  INPUT"COMMAND";C
150  IF C > 2 THEN 115
155  ON INT(C) GOTO 1000, 2000
1000 REM ------------------------------------------------------------------------
1005 REM  ENTER NAMES ROUTINE
1010 REM ------------------------------------------------------------------------
1015 L=L+1
1020 IF L > 100 THEN 1400
1025 PRINT"IF YOU WISH TO EXIT THIS ROUTINE . . . "
1030 PRINT"TYPE ONE OF THESE '#',THEN 'RETURN'."
1035 FOR N = L TO 100 :PRINT:PRINT"NUMBER ";N:PRINT
1040 INPUT"NAME :            ";NA$(N)
1045 IF NA$(N) = "#" THEN 1300
1047 IF NA$(N) = "\" THEN N=N-2 :GOTO 1100
1050 REM
1055 INPUT"COMPANY :         ";CO$(N)
1060 REM
1065 INPUT"ADDRESS :         ";AD$(N)
1070 REM
1075 INPUT"CITY & STATE :    ";CS$(N)
1080 REM
1085 INPUT"ZIP :             ";ZP$(N)
1100 NEXT
1200 L = 100 : GOTO 1500
1300 L = N - 1 : GOTO 1600
1400 L = L - 1
1500 PRINT:PRINT"THE LIST IS FULL.":PRINT
1600 PRINT:PRINT"YOU HAVE ";L;" NAMES ON THIS LIST."
1700 GOTO 100
2000 REM ------------------------------------------------------------------------
2005 REM  PRINT-OUT ROUTINE
2010 REM ------------------------------------------------------------------------
2015 PRINT
2020 PRINT"1) MAKE PAPER TAPE LEADER IN 'LOCAL' MODE."
2025 PRINT
2030 PRINT"2) SWITCH PRINTER TO 'LINE' AND LINE UP LABELS."
2035 PRINT
2040 PRINT"3) TYPE ANY LETTER, THEN 'RETURN'."
2045 PRINT:INPUT"WAITING . . . ";W$
2050 FOR X = 1 TO L STEP 3
2055 Y = X + 1 : Z =X + 2
2060 PRINT TAB(0) ; NA$(X) ; TAB(25) ; NA$(Y) ; TAB(51) ; NA$(Z)
2065 PRINT TAB(0) ; CO$(X) ; TAB(25) ; CO$(Y) ; TAB(51) ; CO$(Z)
2070 PRINT TAB(0) ; AD$(X) ; TAB(25) ; AD$(Y) ; TAB(51) ; AD$(Z)
2075 PRINT TAB(0) ; CS$(X) ; TAB(25) ; CS$(Y) ; TAB(51) ; CS$(Z)
2080 PRINT TAB(0) ; ZP$(X) ; TAB(25) ; ZP$(Y) ; TAB(51) ; ZP$(Z)
2085 PRINT:PRINT
2090 NEXT
2095 GOTO 100
```

Program B. Listing for the bare-bones version of the program. The format is set for a Teletype. Simple adjustments can be made to fit other printers. A paper-tape punch is used to save the list. The Teletype is run in local mode to print additional lists.

This particular method is inexpensive and does not take any time at all to load or make because the paper-tape copy is punched as you print the list! How easy can something be?

If It Works . . . Modify It!

Suppose your names are longer than your labels. When do you abbreviate? Adding the appropriate lines from Fig. 1 allows the computer to count the number of input characters. The LEN function, if you have it (Mits does), can test against the size of your line. If the test is valid, GOSUB to an error message. Further modifications include another module to read a whole letter from cassette using the POKEs given. Then print a personalized copy to each customer on the list.

You can now consider sentence structures like "and in closing, 'Mr. Jones,' we'd like to offer . . . ," just as the big mail-order operations do it! Still another useful modification is a cassette tape directory of your lists . . . a good idea when you get up to a thousand. An excellent example of this method appeared in *73 Magazine* ("The Soft Art of Programming," Parts 1-3, Oct-Dec 1976, by Rich Didday) and was reprinted in *The New Hobby Computers*, 73 Inc., 1977.

We don't have to confine our list to names and addresses. Adding a few more variables in the program allows the luxury of obtaining other important data from our list, such as types of merchandise each customer wants or has ordered. You might choose to save important dates for each customer—write a simple routine to search the current list and pop out names that need collection letters, birthday greetings or warranty follow-up letters. The personalized form letter, mentioned before, could be printed just for those on the list who need it. All you need do is add to the routines given.

Perhaps you can begin to see that what started as a simple system could easily be expanded into a first-class data base for your business. You can start with the program given and upgrade from there, even to disk. You lose nothing by starting now with just the list. In fact, you may gain in the long run because you will be able to tailor the program to your own needs. The really important procedures will be yours, thereby ending forever that locked-in feeling you get with someone else's software.

If you know that feeling or need an upward compatible mailing program for your business, you should get this program up and running and begin to save time and money *now* while planning for the future.■

Sources For More Information

High-speed cassette interface.	Tarbell Electronics 20620 S. Leapwood Ave. Suite P Carson CA 90746 (213) 538-4251
Teletype labels and ready-made forms for printers.	Uarco Incorporated 2600 Wilshire Bl. Suite 408 Los Angeles CA 90057 (213) 380-2595
Custom labels for any printer.	Avery Label Company 777 E Foothill Bl. Azusa CA 91704 (213) 969-3311
Teletype sprocket feed kits and special type fonts. Also carbon ribbons for Teletype.	TTS 2928 Nebraska Ave. Santa Monica CA 90404 (213) 829-2611

Dr. Adam Osborne
Osborne & Associates, Inc.
PO Box 2036
Berkeley CA 94702

Number Crunching: Two Hardware Solutions

faster and smoother than software

People attempting to use microprocessors in scientific applications are probably the first to discover that microprocessors do indeed have limitations. A microprocessor's ability to execute instructions in microseconds may, on the surface, sound very impressive, and it is—until you try to handle trigonometric functions, logarithms, exponentiation or even multidigit multiplication and division.

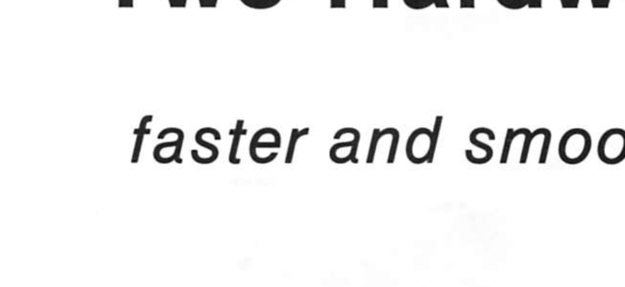


Fig. 1. An arithmetic processor in a microcomputer system.

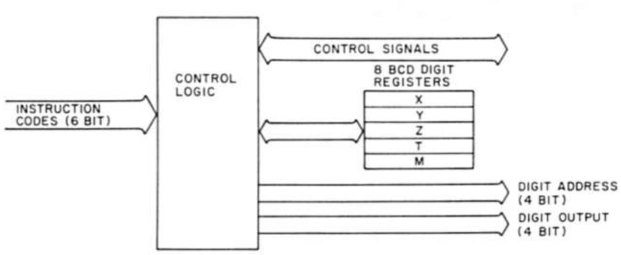


Fig. 2. MM57109 arithmetic processor functional logic.

Trigonometric and logarithmic functions are generally referred to as "transcendental" functions. Writing a microcomputer program to handle transcendental functions is far more difficult than the most complex payroll system could ever be. In fact, designing a program that will generate truly accurate transcendental functions is a formidable task. The problem with these functions is that over limited ranges they change rapidly. Programs that generate transcendental functions must generate very accurate answers, particularly in the fast-moving range, because on rare occasions you will want to subtract almost identical values—and a small difference between two large, erroneous numbers may be completely wrong.

Therefore, when examining arithmetic processors, you must look at the accuracy of results in the fast-moving numeric range. If you are accustomed to evaluating chips simply on the basis of cycle times and programmable options, you now have an important new consideration—the method used to generate results.

Two arithmetic processors will soon be available: the MM57109 from National Semiconductor and the AM9511 from Advanced Micro Devices. About the only thing these two devices have in common is that they both perform approximately the same transcendental functions, and each is treated as a support device within a microcomputer system.

Suppose, for example, you want to compute the natural logarithm of a number. You will transmit the number, as data, to an arithmetic processor, addressing it as an I/O port. At some later time you will read back the answer, as data being input from an I/O port. This use of an arithmetic processor is illustrated conceptually in Fig. 1.

The primary difference between the MM57109 and the AM9511 is that National Semiconductor's MM57109 is a calculator chip; it looks nothing

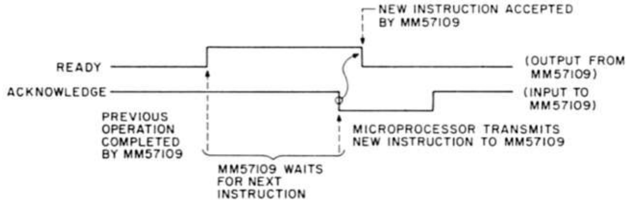


Fig. 3.

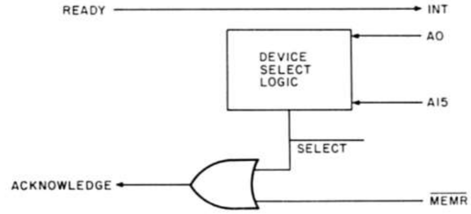


Fig. 4.

like the typical microprocessor support device. The AM9511, in contrast, is immediately recognizable to any experienced microcomputer user as a typical microprocessor support device.

Let's look at each part in turn. The discussion that follows will give you some idea of part capabilities; however, detailed operating procedures are not provided.

MM57109

Fig. 2 illustrates the general logic organization of the MM57109. The most important characteristic of this part is that it operates on binary-coded decimal (BCD) numbers up to eight digits long. Numbers may be handled in fixed-point or floating-point format. A fixed-point number is eight digits long, with a decimal point located at any digit boundary. Thus, numbers in the range 99999999 through .00000001 may be represented.

Floating-point numbers have the form:

(± 0.XXXXXXXX)ExP(± YY)

X and Y represent any decimal digits. Thus, any number in the range $1 \times 10^{+99}$ through 1×10^{-99} can be represented, with eight digits of accuracy.

As you might expect, you must operate the MM57109 by transmitting data and commands to it. Results are received as data. Commands are summarized in Table 1. Note that the MM57109 is not a fast device. Execution times are shown based on a ten-microsecond microcycle, the recommended maximum rate for this device. It takes at least four milliseconds to enter a single eight-digit number (in fixed- or floating-point notation), while trigonometric functions may take almost a second to resolve.

In order to cope with these relatively slow times, all data communications between the MM57109 and a microprocessor use request/acknowledge handshaking control signal protocol. Upon completing any operation, the MM57109 outputs a ready signal true. Normally the microprocessor will hold an acknowledge input false to suppress any new operations. Upon detecting the true ready, the microprocessor will transmit a new command to the MM57109 and set the acknowledge input true. This is illustrated in Fig. 3.

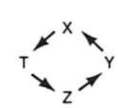
This handshaking scheme readily lends itself to almost any microprocessor; the ready "true" signal can be used to request an interrupt, while the acknowledge can be tied directly to a combined MM57109 device-select and write-control signal. For 8080A signals, this is illustrated in Fig. 4.

The method of transmitting control commands to an MM57109 device differs markedly from the standard method used within microcomputer systems. The standard method (which is used by the AM9511) takes the device-select logic output to a select pin, then has a control/data discriminator that usually constitutes part of the device address. Memory-read and memory-write control signals then become simple control strobes that accompany an address-activated select logic. Fig. 5 illustrates this.

There are three ways you can enter data to the MM57109; in each case the register stack is pushed and data is written into the X register (see Fig. 6).

The first data entry method is approximately equivalent to calculator-keyboard entry; separate commands identify the

*Table 1. MM57109 instruction description table (*indicates two-word instruction).*

MNEMONIC*	EXECUTION TIME (MICROCYCLES) (AVERAGE)	EXECUTION TIME (MICROCYCLES) (WORST-CASE VALUES)	DESCRIPTION
0		238	Mantissa or exponent digits. On first digit (d) the
1		238	following occurs: Z→T
2		238	Y→Z
3		238	X→Y
4		238	d→X
5		238	See description of number entry on page 11.
6		238	
7		238	
8		238	
9		238	
DP		152	Digits that follow will be mantissa fraction.
EE		151	Digits that follow will be exponent.
CS		166	Change sign of exponent or mantissa. Xm = X mantissa Xe = X exponent CS causes −Xm→Xm or −Xe→Xe depending on whether or not an EE instruction was executed after last number entry initiation.
PI		1312	3.1415927→X, stack not pushed.
EN		552	Terminates digit entry and pushes the stack. The argument entered will be in X and Y. Z→T Y→Z X→Y
NOP		122	Do nothing instruction that will terminate digit entry.
HALT		134	External hardware detects HALT op code and generates HOLD = 1. Processor waits for HOLD = 0 before continuing. HALT acts as a NOP and may be inserted between digit entry instructions since it does not terminate digit entry.
ROLL		905	Roll Stack. 

POP		448	Pop Stack. $Y \rightarrow X$ $Z \rightarrow Y$ $T \rightarrow Z$ $O \rightarrow T$
XEY		652	Exchange X and Y. $X \leftrightarrow Y$
XEM		812	Exchange X with memory. $X \leftrightarrow M$
MS		839	Store X in Memory. $X \rightarrow M$
MR		1385	Recall Memory into X. $M \rightarrow X$
LSH		168	X mantissa is left shifted while leaving decimal point in same position. Former most significant digit is saved in link digit. Least significant digit is zero.
RSH		173	X mantissa is right shifted while leaving decimal point in same position. Link digit, which is normally zero except after a left shift, is shifted into the most significant digit. Least significant digit is lost.
+	2200	6600	Add X to Y. $X + Y \rightarrow X$. On +, −, ×, / and YX instructions, stack is popped as follows: $Z \rightarrow Y$ $T \rightarrow Z$ $O \rightarrow T$ Former X, Y are lost.
−	2200	6600	Subtract X from Y. $Y - X \rightarrow X$
×	3200	22700	Multiply X times Y. $Y \times X \rightarrow X$
/	7800	22300	Divide X into Y. $Y \div X \rightarrow X$
YX	55400	95500	Raise Y to X power. $Y^X \rightarrow X$
INV + *	1700	5000	Add X to memory. $M + X \rightarrow M$ On INV +, −, × and / instructions, X, Y, Z, and T are unchanged.
INV − *	1700	5000	Subtract X from memory. $M - X \rightarrow M$
INV × *	2700	21400	Multiply X times memory. $M \times X \rightarrow M$
INV/*	7300	21100	Divide X into memory. $M \div X \rightarrow M$
1/X	4500	22800	$1 \div X \rightarrow X$. On all F (X) math instructions Y,Z,T and M are unchanged and previous X is lost.
SQRT	7000	30200	$\sqrt{X} \rightarrow X$
SQ	3000	21900	$X^2 \rightarrow X$
10X	27400	96500	$10^X \rightarrow X$
EX	30800	93900	$e^X \rightarrow X$
LN	24800	92000	$\ln X \rightarrow X$
LOG	30700	92600	$\log X \rightarrow X$
SIN	56200	95900	$SIN(X) \rightarrow X$. On all F(X) trig functions, Y,Z,T, and M are unchanged and the previous X is lost.
COS	56200	95900	$COS(X) \rightarrow X$
TAN	35000	97600	$TAN(X) \rightarrow X$
INV SIN*	54000	93900	$SIN^{-1}(X) \rightarrow X$
INV COS*	54000	93900	$COS^{-1}(X) \rightarrow X$
INV TAN*	30200	92900	$TAN^{-1}(X) \rightarrow X$
DTR	9600	41700	Convert X from degrees to radians.
RTD	9600	41700	Convert X from radians to degrees.
MCLR		734	Clear all internal registers and memory; initialize I/O control signals, MDC = 8, MODE = floating point. (See initialization.)
ECLR		163	$O \rightarrow$ Error flag
JMP*		186	Unconditional branch to address specified by second instruction word. On all branch instructions, second word contains branch address to be loaded into external PC.
TJC*		208	Branch to address specified by second instruction word if JC (I_6) is true (= 1). Otherwise, skip over second word.
TERR*		191	Branch to address specified by second instruction word if error flag is true (= 1). Otherwise, skip over second word. May be used for detecting specific errors as opposed to using the automatic error recovery scheme dealt with in the section on Error Control.
TX=0*		278	Branch to address specified by second instruction word if X = 0. Otherwise, skip over second word.
TXF*		277	Branch to address specified by second instruction word if $\|X\| < 1$. Otherwise, skip over second word. (i.e., branch if X is a fraction.)
TXLT0*		197	Branch to address specified by second instruction word if X < 0. Otherwise, skip over second word.
IBNZ		2314	$M + 1 \rightarrow M$. If M = 0, skip second instruction word. Otherwise, branch to address specified by second instruction word.
DBNZ		2314	$M - 1 \rightarrow M$. If M = 0, skip second instruction word. Otherwise, branch to address specified by second instruction word.
IN*		395	The processor supplies a 4-bit digit address ($\overline{DA4}$-$\overline{DA1}$) accompanied by a digit address strobe ($\overline{DAS}$)

decimal digits 0 through 9, the decimal point and signs for the mantissa and exponent—if floating-point format is specified.

The other two input techniques transmit data to the X register under program control. An IN instruction is executed once for entry of an entire number, while an AIN instruction is executed once per digit of a number being entered. In each case the number is entered into the X register after the stack is pushed, as illustrated for keyboard entry. Following execution for the IN or AIN instruction, digits are entered as data. Input is clocked by an output control signal accompanying the 4-bit digit address illustrated in Fig. 2.

Handshaking protocol similar to the ready-acknowledge sequence illustrated for instruction input controls data entry. Thus, it is relatively easy for any microprocessor to work asynchronously with the MM57109.

MM57109 data output is controlled by an OUT instruction which is equivalent to the IN instruction.

MM57109 data input and output philosophy contrast sharply with normal microprocessor protocol. Observe that the MM57109 requires the microprocessor to input an appropriate control command, after which the MM57109 outputs strobe signals to time data input or output. Thus, the MM57109 is not behaving like a standard peripheral device, rather, it becomes temporary bus master while inputting or outputting data.

In a normal microcomputer system, the microprocessor will input or output data from a support device just as it would for read/write memory. The device is selected via an appropriate I/O port or memory address, then a read or write control signal causes the data transfer to occur; this is how the AM9511 works.

National Semiconductor literature describes the MM57109 as either a stand-alone microprocessor or as an adjunct to

		for each digit to be input. The high order address for the number to be input would typically come from the second instruction word. The digit is input on D4-D1, using ISEL = 0 to select digit data instead of instructions. The number of digits to be input depends on the calculation mode (scientific notation or floating point) and the mantissa digit count (see Data Formats and Instruction Timing). Data to be input is stored in X and the stack is pushed (X → Y → Z → T). At the conclusion of the input, DA4-DA1 = 0.
OUT*	583	Addressing and number of digits is identical to IN instruction. Each time a new digit address is supplied, the processor places the digit to be output on DO4-DO1 and pulses the R/$\overline{W}$ line active low. At the conclusion of output, DO4-DO1 = 0 and DA4-DA1 = 0.
AIN	284	A single digit is read into the processor on D4-D1. ISEL = 0 is used by external hardware to select the digit instead of instruction. It will not read the digit until $\overline{ADR}$ = 0 (ISEL = 0 selects $\overline{ADR}$ instead of I_5), indicating data valid. F2 is pulsed active low to acknowledge data just read.
SF1	163	Set F1 high, i.e., F1 = 1.
PF1	185	F1 is pulsed active high. If F1 is already high, this results in it being set low.
SF2	163	Set F2 high, i.e., F2 = 1.
PF2	185	F2 is pulsed active high. If F2 is already high, this results in it being set low.
PRW1	130	Generates R/$\overline{W}$ active low pulse which may be used as a strobe or to clock extra instruction bits into a flip-flop or register.
PRW2	130	Identical to PRW1 instruction. Advantage may be taken of the fact that the last 2 bits of the PRW1 op code are 10 and the last 2 bits of the PRW2 op code are 01. Either of these bits can be clocked into a flip-flop using the R/$\overline{W}$ pulse.
TOGM	157	Change mode from floating point to scientific notation or vice versa, depending on present mode. The mode affects only the IN and OUT instructions. Internal calculations are always in 8-digit scientific notation.
SMDC*	163	Mantissa digit count is set to the contents of the second instruction word (= 1 to 8).
INV	166	Set inverse mode for trig or memory function instruction that will immediately follow. Inverse mode is for next instruction only.

another microprocessor. In reality, the MM57109 is not a practical stand-alone microprocessor. It should be used only in conjunction with another microprocessor because the MM57109 has no internal memory-addressing logic. A program counter, if present, must be implemented externally, using some appropriate register whose contents get triggered when appropriate timing signals are output by the MM57109. Branch instructions, though identified in Table 1, really do not exist; they simply create a control signal that external logic must use to clock an address into the external program counter.

By the time you have configured the necessary additional logic to surround a stand-alone MM57109, you will probably find it is cheaper and a good deal faster to use some simple microprocessor, even if its sole function is to monitor and control MM57109 operations.

AM9511

Now let's look at the AM9511. Functional logic for this device is illustrated in Fig. 7. The most important difference between the AM9511 and the MM57109 is that the AM9511 is a binary device. All data operations within the AM9511 handle binary data; in contrast, the MM57109 handles only BCD data. AM9511 data may be specified in fixed-point or floating-point format. Fixed-point numbers may be single- or double-precision; in each case they are treated as signed binary numbers. A single-precision fixed-point number is illustrated in Fig. 8.

This is standard signed binary data. Thus, single-precision fixed-point numbers may range in value from −32768 to +32767. Double-precision fixed-point numbers are 32 bits wide, and again use standard signed binary data format. Thus, a double-precision number may have values in the range −2147483648 through +2147483647.

Floating-point numbers are all 32 bits wide, and are interpreted as in Fig. 9. The mantissa and exponent are both binary numbers; therefore, numbers in the range $\pm(2.7 \times 10^{-20}$ to $9.2 \times 10^{-18})$ may be represented.

Observe that the AM9511 has a smaller range of valid numbers than the MM57109. You might argue that the AM9511, by handling numbers in the exponential range 10^{-20} through 10^{18}, must surely have a range adequate for any application. This is not always true.

In particular, chemical-engineering and astronomical computations frequently handle numbers outside the range allowed by the AM9511. The principal advantage of the AM9511 over the MM57109 is that the former is much faster. Table 2 summarizes AM9511 instructions. Notice that the instruction sets for the two devices are approximately

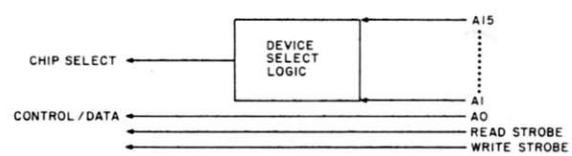


Fig. 5.

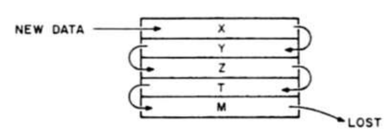


Fig. 6.

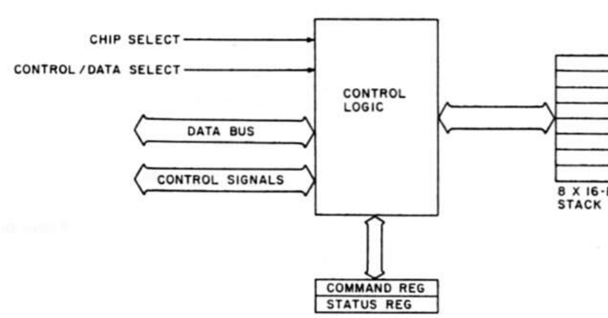


Fig. 7. AM9511 arithmetic processor functional logic.

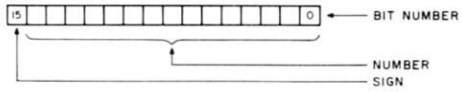


Fig. 8. A single-precision fixed-point number.

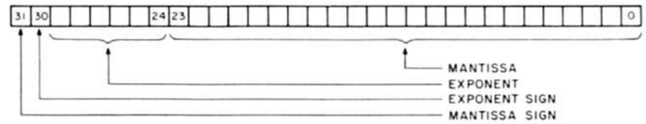


Fig. 9.

equivalent; however, based on a 500-nanosecond clock, for the AM9511 it is more than 100 times faster than for the MM57109. Also, the AM9511 is incredibly easy to incorporate into almost any microcomputer system. Control signals, data buses and address buses are typical of an 8080A support device. The AM9511 is selected via the chip-select (CS) and $C/\overline{D}$ inputs. This is the standard method used in any 8080A support device to access data control and status locations as two memory addresses or I/O ports.

The standard read and write control strobes are used to input or output data. Thus, the CS, $C/\overline{D}$, $\overline{RD}$ and $\overline{WR}$ controls together identify events as in Table 3.

Data and instructions are input via the bidirectional data bus; results and status are output via the same bus. While the AM9511 is busy executing any operation, a PAUSE signal is output low. At the end of the operation the END control signal is output low. The microprocessor acknowledges the END output by inputting EACK low.

CS	$C/\overline{D}$	$\overline{RD}$	$\overline{WR}$	Function
1	X	X	X	Device not selected
0	0	0	1	Read data from device
0	0	1	0	Write data to device
0	1	0	1	Read status from device
0	1	1	0	Write command to device

Table 3.

Any command output to the AM9511 can, in addition to all other options, specify a service request to follow completion of the AM9511 operation. During a service request, CPU will process AM9511 results before initiating a new AM9511 operation. If a service request is specified, when the AM9511 completes any operation it outputs a low service-request signal. The CPU acknowledges this signal with a service-acknowledge input. Thus, the AM9511 allows the microprocessor to differentiate between an AM9511 operation that does or does not require further handling by the CPU.

When you compare the AM9511 and MM57109 devices, selection should be based on the following trade-offs:

1. The MM57109 is a BCD device and will therefore be easier to use in a purely decimal application.

2. The MM57109 has a larger numeric range; however, you should be sure that the extensive AM9511 numeric range is insufficient before you go to the MM57109 based upon this criterion.

3. The AM9511 is significantly faster than the MM57109. There may be applications in which the AM9511 must be selected based on its speed, even if BCD-to-binary and binary-to-BCD conversions are required.

4. The AM9511 fits naturally into any 8080A microcomputer configuration; its bus and control signal interface is absolutely compatible with the 8080A. In contrast, the MM57109 is a calculator part that will need multiplexing and de-multiplexing circuits surrounding it.

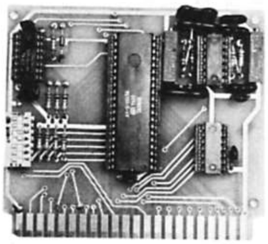
Whether you choose the AM9511 or the MM57109, you will be making the right choice if your alternative is to write your own transcendental-function calculations. ■

COMMAND MNEMONIC	CLOCK CYCLES	COMMAND DESCRIPTION (1)
SADD	17	Adds TOS to NOS. Result to NOS. Pop Stack
SSUB	30	Subtracts TOS from NOS. Result to NOS. Pop Stack
SMUL	92	Multiplies NOS by TOS. Result to NOS. Pop Stack
SDIV	92	Divides NOS by TOS. Result to NOS. Pop Stack
DADD	21	Adds TOS to NOS. Result to NOS. Pop Stack
DSUB	38	Subtracts TOS from NOS. Result to NOS. Pop Stack
DMUL	208	Multiplies NOS by TOS. Result to NOS. Pop Stack
DDIV	208	Divides NOS by TOS. Result to NOS. Pop Stack
FADD	56-350	Adds TOS to NOS. Result to NOS. Pop Stack
FSUB	58-352	Subtracts TOS from NOS. Result to NOS. Pop Stack
FMUL	168	Multiplies NOS by TOS. Result to NOS. Pop Stack
FDIV	171	Divides NOS by TOS. Result to NOS. Pop Stack
SQRT	800	Square Root of TOS. Result in TOS.
SIN	4464	Sine of TOS. Result in TOS.
COS	4118	Cosine of TOS. Result in TOS.
TAN	5754	Tangent of TOS. Result in TOS.
ASIN	7668	Inverse Sine of TOS. Result in TOS.
ACOS	7734	Inverse Cosine of TOS. Result in TOS.
ATAN	6006	Inverse Tangent of TOS. Result in TOS.
LOG	4490	Common Logarithm (base 10) or TOS. Result in TOS.
LN	4478	Natural Logarithm (base e) of TOS. Result in TOS.
EXP	4616	Exponential (e^x) of TOS. Result in TOS.
PWR	9292	NOS raised to the power in TOS. Result to NOS. Pop Stack.
NOP	4	No Operation
FIXS	92-216	Converts TOS from floating-point to single-precision fixed-point format.
FIXD	100-346	Converts TOS from floating-point to double-precision fixed-point format.
FLTS	98-186	Converts TOS from single-precision fixed-point to floating-point format.
FLTD	98-378	Converts TOS from double-precision fixed-point to floating-point format.
CHSS	26	Changes sign of single-precision fixed-point operand on TOS.
CHSD	34	Changes sign of double-precision fixed-point operand on TOS.
CHSF	16	Changes sign of floating-point operand on TOS.
PTOS	16	Push single-precision fixed-point operand on TOS to NOS.
PTOD	20	Push double-precision fixed-point operand on TOS to NOS.
PTOF	20	Push floating-point operand on TOS to NOS.
POPS	10	Pop single-precision fixed-point operand from TOS. NOS becomes TOS.
POPD	12	Pop double-precision fixed-point operand from TOS. NOS becomes TOS.
POPF	12	Pop floating-point operand from TOS. NOS becomes TOS.
XCHS	18	Exchange single-precision fixed-point operands TOS and NOS.
XCHD	26	Exchange double-precision fixed-point operands TOS and NOS.
XCHF	26	Exchange floating-point operands TOS and NOS.
PUPI	16	Push floating-point constant "π" onto TOS. Previous TOS becomes NOS.

Notes: 1. *Nomenclature:* TOS is Top Of Stack. NOS is Next On Stack.
2. All derived floating-point functions destroy the contents of the stack. Only the result can be counted on the be valid upon command completion.
3. Format conversion commands (FIXS, FIXD, FLTS, FLTD) require that floating-point data format be specified (command bits 5 and 6 must be 0).

Table 2. AM9511 instruction description table.

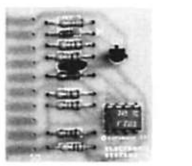
ELECTRONIC SYSTEMS
P.O. Box 9641 San Jose CA 95157
(408) 374-5984



UART & BAUD RATE GENERATOR

Part no. 101
- Converts serial to parallel and parallel to serial
- Low cost on board baud rate generator
- Baud rates: 110, 150, 300, 600, 1200, and 2400
- Low power drain +5 volts and -12 volts required
- TTL compatible
- All characters contain a start bit, 5 to 8 data bits, 1 or 2 stop bits, and either odd or even parity.
- All connections go to a 44 pin gold plated edge connector
- Board only $12.00; with parts $35.00

RS-232/TTL INTERFACE

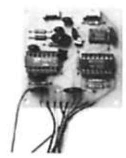


Part no. 232
- Converts TTL to RS-232, and converts RS-232 to TTL
- Two separate circuits
- Requires -12 and +12 volts
- All connections go to a 10 pin gold plated edge connector
- Board only $4.50; with parts $7.00

DC POWER SUPPLY

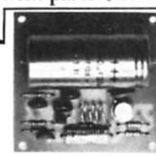
Part no. 6085
- Board supplies a regulated +5 volts at 3 amps., +12,-12, and -5 volts at 1 amp.
- Circuit has filters, rectifiers, and regulators.
- Power required is 8 volts AC at 3 amps., and 24 volts AC C.T. at 1.5 amps.
- Board only $12.50

Part no. 111



TAPE INTERFACE

- Play and record Kansas City Standard tapes
- Converts a low cost tape recorder to a digital recorder
- Works up to 1200 baud
- Digital in and out are TTL-serial
- Output of board connects to mic. in of recorder
- Earphone of recorder connects to input on board
- Requires +5 volts, low power drain
- Board $7.60; with parts $27.50
- No coils

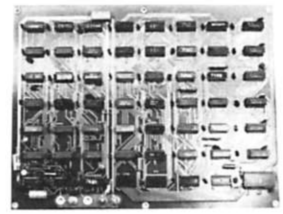


Part no. 107

RF MODULATOR

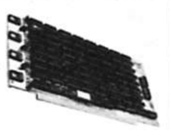
- Converts video to AM modulated RF, Channels 2 or 3
- Power required is 12 volts AC C.T., or +5 volts DC
- Board $7.60; with parts $13.50

TELEVISION TYPEWRITER



Part no. 106
- Stand alone TVT
- 32 char/line, 16 lines, modifications for 64 char/line included
- Parallel ASCII (TTL) input
- Video output
- 1K on board memory
- Output for computer controlled curser
- Auto scroll
- Non-distructive curser
- Curser inputs: up, down, left, right, home, EOL, EOS
- Scroll up, down
- Requires +5 volts at 1.5 amps, and -12 volts at 30 mA
- Board only $39.00; with parts $145.00

8K STATIC RAM



Part no. 300
- 8K Altair bus memory
- Uses 2102 Static memory chips
- Memory protect
- Gold contacts
- Wait states
- On board regulator
- S-100 bus compatible
- Vector input option
- TRI state buffered
- Board only $22.50; with parts $160.00

TIDMA

Part no. 112
- Tape Interface Direct Memory Access
- Record and play programs without bootstrap loader (no prom) has FSK encoder/decoder for direct connections to low cost recorder at 625 baud rate, and direct connections for inputs and outputs to a digital recorder at any baud rate.
- S-100 bus compatible
- Board only $35.00; with parts $110.00

Apple II Serial I/O Interface

Part No. 2
- Baud rates up to 30,000
- Plugs into Apple Peripheral connector
- Low-current drain
- RS-232 Input and Output

SOFTWARE
- Input and Output routine from monitor or BASIC to teletype or other serial printer.
- Program for using an Apple II for a video or an intelligent terminal. Board only – $15.00; with parts – $42.00; assembled and tested – $62.00.

MODEM

Part no. 109
- Type 103
- Full or half duplex
- Works up to 300 baud
- Originate or Answer
- No coils, only low cost components
- TTL input and output-serial
- Connect 8 ohm speaker and crystal mic. directly to board
- Uses XR FSK demodulator
- Requires +5 volts
- Board $7.60; with parts $27.50

To Order:

Mention part number and description. For parts kits add "A" to part number. Shipping paid for orders accompanied by check, money order, or Master Charge, BankAmericard, or VISA number, expiration date and signature. Shipping charges added to C.O.D. orders. California residents add 6.5% for tax. Parts kits include sockets for all ICs, components, and circuit board. Documentation is included with all products. Dealer inquiries invited. 24 Hour Order Line: (408) 374-5984.

VISA master charge

E21

Ken Barbier
Borrego Engineering
PO Box 1253
Borrego Springs CA 92004

Money Manipulations

keep ahead of those cash-flow problems

```
1 REM    CASH FLOW PROGRAM        29 JULY 77
7 A$="$$#######.##"
8 B$="#####"
9 PRINT:PRINT:PRINT
10 PRINT"INVESTMENT MINUS DRAW"
11 PRINT"                    QUARTERLY STATEMENT"
15 PRINT
20 PRINT"PRINCIPAL: $; INTEREST: %/YR; DRAW: $/MO."
21 PRINT"  INTEREST EARNED SHOWN BY QUARTER TOTAL"
22 PRINT" DRAW IS CURRENT MONTHLY RATE, INFLATED"
25 PRINT:PRINT
30 M=0
31 Y=0
35 C=2
40 INPUT "PRINCIPAL=";P
50 INPUT "INTEREST=";I
60 INPUT "DRAW=";D
61 INPUT "INFLATION=";A
62 A=A/1200
70 I=I/1200
80 PRINT:PRINT:PRINT
90 PRINT"MONTH  PRINCIPAL  EARNED  DRAW"
100 E=P*I
110 P=P+E
120 P=P-D
125 IF P<0 THEN 190
130 M=M+1
135 D=D*(1+A)
137 Y=Y+E
138 IF M=123 THEN 180
140 IF M>240 THEN 1200
141 IF C=0 GOTO 144
142 C=C-1
143 GOTO 100
144 C=2
146 PRINT USING B$;M::PRINT" ";
150 PRINT USING A$;P,Y,D
151 Y=0
170 GOTO 100
180 PRINT:PRINT:PRINT: PRINT
181 GOTO 141
190 PRINT:PRINT:PRINT:PRINT:PRINT
1200 END
```

Fig. 1. Cash Flow program listing. Written in 12K Extended BASIC, it can be run on smaller BASICs by changing lines 146 and 150 to: 150 PRINT M,P,Y,D. Lines 7 and 8 can then be dropped. The output will not be so nicely formatted, however.

You say you're getting ready to punch your boss's lights out, but you're not sure your life savings will support the wife and kiddies until you get out of jail? Or maybe you're just getting old (like me) and think it's time to retire, but you want to be sure you have enough loot stashed away to supplement Uncle Sam's pittance and provide enough to live on – forever and ever. Or perhaps you are ready to throw in the towel at the boiler factory and open your own computer store . . . and want to know how long you can hold out until the first cash customer comes walking in. Well, tell you what I'm going to do . . .

Computing Cash Flow

The Cash Flow program listing in Fig. 1 assumes that an initial investment is made at a fixed rate of interest (compounded monthly). But instead of simply figuring compound interest, Cash Flow assumes that we will be drawing on these reserves, for reasons such as those listed above. Furthermore, life being what it is, the amount we have to withdraw will be subject to inflation, so the program takes this factor into account as well. Since Uncle Sam insists we pay income tax on the interest paid on our investments, we will also need a statement showing interest earned. While the program will not fill out your income tax form for you, it will, considering all these factors, tell you how long your loot will last.

For example, let's take a look at a typical Cash Flow run (Fig. 2). Dick and Jane have both been working and diligently squirreling money away. They have accumulated forty kilobucks and would like to use it to finance an early retirement. What they need to know is whether or not the money will hold out until social security helps them out (assuming it doesn't go broke first).

Being conservative, they will invest the money in insured savings, which, for our example, we will assume pays 5.75 percent per year, compounded monthly. They have moved into a less expensive house, but there are still payments to make. Now, our couple must figure the maximum amount per month that they will have to draw from their savings to live on. This fictional account shows that they have arrived at a figure of $750 per month, which certainly should be enough to feed two mouths.

Next, we throw in a little magic. D. and J. have consulted their financial expert, and he assures them they can expect an inflation rate of 3 percent per annum to apply to the commodities they will be consuming. This figure sounds low today, but if coffee, new cars, etc., are avoided it is not too unrealistic.

All the above conditions established, we load Cash Flow, which is written in Altair BASIC, 12K Extended, Version 3.2. Instructing it to run, we are informed that we will be provided with a quarterly statement, and we are asked to enter the amount of principal (in dollars); the interest rate (in percent per annum); the amount we wish to withdraw (in dollars per month); and the expected annual rate of inflation. Having received these variables, Cash Flow proceeds to produce the quarterly-statement table shown in Fig. 2.

Since this is a quarterly statement, the number of the month for which the figures apply will increment by three. The amount of principal remaining at the end of that month is shown in the next column. The third column shows the total interest earned for the previous quarter, which is what we will have to pay income tax on. This last column shows our draw for the current month. This amount always increases because we have to assume that inflation will continue to spiral.

When all of the money is used up, Cash Flow will terminate, and we will have to go back to work. We see that Dick and Jane can survive for about five years. Well, maybe they'd better try to cut costs a little. Then we can try the program again, using a lower Draw figure.

When this program was first run, the nice round numbers in the cents column under Principal raised suspicion. The BASIC manual states that single-precision numbers are printed with a maximum of six decimal digits, and we are asking BASIC to work with seven digits! So, we should add the following line to our program: 2 DEFDBL P.

Now when we run the program with the same variables, we get the output shown in Fig. 3, since Principal is computed in double precision. We can see the pennies and nickles, but the results don't change! This is because we had sufficient accuracy to begin with, the internal representation of our principal being in binary bits, which don't exactly relate evenly to six-decimal digits. Our initial accuracy was *barely* sufficient, though, so it would be a good idea to leave the second line in our program, in case a rich uncle dies and leaves more money to play with.

Since Dick and Jane are only 23 years old (surprise!) they have decided to postpone the early retirement and keep on working and saving. Now they can use the same program to estimate how their savings will grow if left untouched. If no money is drawn from the investment, Cash Flow becomes a straightforward compound-interest program, as we can see in Fig. 4.

Here, we set draw and inflation to zero, and Cash Flow gives a quarterly statement of earnings and accumulation for our savings account. The program gets tired and quits after 20 years. Dick and Jane probably will, too! ■

```
INVESTMENT MINUS DRAW
          QUARTERLY STATEMENT

PRINCIPAL:   $;   INTEREST: %/YR;   DRAW:  $/MO.
   INTEREST EARNED SHOWN BY QUARTER TOTAL
   DRAW IS CURRENT MONTHLY RATE, INFLATED

PRINCIPAL=? 40000
INTEREST=? 5.75
DRAW=? 750
INFLATION=? 3

MONTH      PRINCIPAL      EARNED      DRAW
  3        $38311.30      $566.95     $755.64
  6        $36581.20      $542.48     $761.32
  9        $34808.90      $517.41     $767.05
 12        $32993.80      $491.73     $772.81
 15        $31135.00      $465.43     $778.62
 18        $29231.70      $438.49     $784.48
 21        $27283.30      $410.92     $790.38
 24        $25289.00      $382.69     $796.32
 27        $23247.80      $353.80     $802.31
 30        $21159.10      $324.23     $808.34
 33        $19022.00      $293.97     $814.42
 36        $16835.70      $263.02     $820.54
 39        $14599.20      $231.35     $826.71
 42        $12311.90      $198.96     $832.93
 45         $9972.66      $165.83     $839.19
 48         $7580.74      $131.95     $845.50
 51         $5135.21       $97.31     $851.86
 54         $2635.15       $61.90     $858.26
 57           $79.62       $25.69     $864.71
```

Fig. 2. Sample Cash Flow run. This printout shows how long an initial investment of $40,000 will last while earning 5.75 percent interest, but being drawn on at the rate of $750 per month, inflated 3 percent per year.

```
INVESTMENT MINUS DRAW
          QUARTERLY STATEMENT

PRINCIPAL:   $;   INTEREST: %/YR;   DRAW:  $/MO.
   INTEREST EARNED SHOWN BY QUARTER TOTAL
   DRAW IS CURRENT MONTHLY RATE, INFLATED

PRINCIPAL=?  40000
INTEREST=?   5.75
DRAW=? 750
INFLATION=? 3

MONTH      PRINCIPAL      EARNED      DRAW
  3        $38311.32      $566.95     $755.64
  6        $36581.21      $542.48     $761.32
  9        $34808.94      $517.41     $767.05
 12        $32993.78      $491.73     $772.81
 15        $31134.96      $465.43     $778.62
 18        $29231.74      $438.49     $784.48
 21        $27283.34      $410.92     $790.38
 24        $25288.97      $382.69     $796.32
 27        $23247.83      $353.80     $802.31
 30        $21159.12      $324.23     $808.34
 33        $19022.01      $293.97     $814.42
 36        $16835.66      $263.02     $820.54
 39        $14599.24      $231.35     $826.71
 42        $12311.86      $198.96     $832.93
 45         $9972.66      $165.83     $839.19
 48         $7580.75      $131.95     $845.50
 51         $5135.22       $97.31     $851.86
 54         $2635.15       $61.90     $858.26
 57           $79.62       $25.69     $864.71
```

Fig. 3. A double-precision run. The net results have not changed, but would for larger principals. Double precision results in a more accurate printout, but the program takes longer to run.

```
INVESTMENT MINUS DRAW
          QUARTERLY STATEMENT

PRINCIPAL:   $;   INTEREST:   %/YR;   DRAW:  $/MO.
   INTEREST EARNED SHOWN BY QUARTER TOTAL
   DRAW IS CURRENT MONTHLY RATE, INFLATED

PRINCIPAL=? 10000
INTEREST=? 6
```

DRAW=? 0
INFLATION=? 0

MONTH	PRINCIPAL	EARNED	DRAW
3	$10150.75	$150.75	$0.00
6	$10303.78	$153.02	$0.00
9	$10459.11	$155.33	$0.00
12	$10616.78	$157.67	$0.00
15	$10776.83	$160.05	$0.00
18	$10939.29	$162.46	$0.00
21	$11104.20	$164.91	$0.00
24	$11271.60	$167.40	$0.00
27	$11441.52	$169.92	$0.00
30	$11614.00	$172.48	$0.00
33	$11789.08	$175.08	$0.00
36	$11966.81	$177.72	$0.00
39	$12147.21	$180.40	$0.00
42	$12330.33	$183.12	$0.00
45	$12516.21	$185.88	$0.00
48	$12704.89	$188.68	$0.00
51	$12896.42	$191.53	$0.00
54	$13090.83	$194.42	$0.00
57	$13288.18	$197.35	$0.00
60	$13488.50	$200.32	$0.00
63	$13691.84	$203.34	$0.00
66	$13898.25	$206.41	$0.00
69	$14107.77	$209.52	$0.00
72	$14320.44	$212.68	$0.00
75	$14536.33	$215.88	$0.00
78	$14755.46	$219.14	$0.00
81	$14977.90	$222.44	$0.00
84	$15203.70	$225.79	$0.00
87	$15432.89	$229.20	$0.00
90	$15665.55	$232.65	$0.00
93	$15901.71	$236.16	$0.00
96	$16141.43	$239.72	$0.00
99	$16384.76	$243.33	$0.00
102	$16631.76	$247.00	$0.00
105	$16882.49	$250.73	$0.00
108	$17136.99	$254.51	$0.00
111	$17395.34	$258.34	$0.00
114	$17657.57	$262.24	$0.00
117	$17923.76	$266.19	$0.00
120	$18193.97	$270.20	$0.00
123	$18468.24	$274.28	$0.00
126	$18746.65	$278.41	$0.00
129	$19029.26	$282.61	$0.00
132	$19316.13	$286.87	$0.00
135	$19607.32	$291.19	$0.00
138	$19902.91	$295.58	$0.00
141	$20202.95	$300.04	$0.00
144	$20507.51	$304.56	$0.00
147	$20816.66	$309.15	$0.00
150	$21130.47	$313.81	$0.00
153	$21449.02	$318.54	$0.00
156	$21772.37	$323.35	$0.00
159	$22100.59	$328.22	$0.00
162	$22433.76	$333.17	$0.00
165	$22771.95	$338.19	$0.00
168	$23115.24	$343.29	$0.00
171	$23463.70	$348.47	$0.00
174	$23817.42	$353.72	$0.00
177	$24176.47	$359.05	$0.00
180	$24540.94	$364.46	$0.00
183	$24910.89	$369.96	$0.00
186	$25286.43	$375.53	$0.00
189	$25667.62	$381.20	$0.00
192	$26054.57	$386.94	$0.00
195	$26447.34	$392.78	$0.00
198	$26846.04	$398.70	$0.00
201	$27250.75	$404.71	$0.00
204	$27661.55	$410.81	$0.00
207	$28078.56	$417.00	$0.00
210	$28501.84	$423.29	$0.00
213	$28931.51	$429.67	$0.00
216	$29367.66	$436.15	$0.00
219	$29810.38	$442.72	$0.00
222	$30259.78	$449.40	$0.00
225	$30715.95	$456.17	$0.00
228	$31178.99	$463.05	$0.00
231	$31649.02	$470.03	$0.00
234	$32126.13	$477.11	$0.00
237	$32610.44	$484.31	$0.00
240	$33102.04	$491.61	$0.00

Fig. 4. Compound-interest run. If Draw is set to zero, Cash Flow becomes a straight compound-interest computation. Here, $10,000 was invested at 6 percent for 20 years. Changing program line 140 can vary this time limit.

THE MICRO WORKS

COME ON UP AND SEE ME SOMETIME

Give your 6800 computer the gift of sight! The Micro Works Digisector® opens up a whole new world for your computer. Your micro can now be a part of the action, taking pictures like this one to amuse your friends, watching your home while you're away, helping your household robot avoid bumping into walls, providing fast to slow scan conversion for you hams . . . the applications abound.

The Micro Works Digisector is a completely unique device; its resolution and speed are unmatched in industry and the price is unbeatable anywhere. The Digisector and a cheap TV camera are all you'll need to see eye to eye with your 6800. Since operation is straightforward, you don't have to be a software wizard to utilize the Digisector's extensive capabilities. The Micro Works Digisector board provides the following exclusive features:

- High Resolution—a 256 x 256 picture element scan
- Precision—64 levels of grey scale
- Speed—Conversion times as low as 3 microseconds per pixel
- Versatility—Accepts either interlaced (NTSC) or non-interlaced (Industrial) video input
- Compactness—Utilizes 1 I/O slot in your SWTPC 6800 or equivalent
- Economy—The Digisector is a professional tool priced for the hobbyist

The Digisector (DS-68), like all Micro Works products, comes fully assembled, tested and burned in. Only the highest quality components are used, and the boards are double sided with plated through holes, solder mask and silkscreen. All software is fully source listed and commented. The Micro Works is proud to add the DS-68 to its line of quality computer accessories for the hobbyist. **Price 169.95**

The Micro Works 6800 series of computer accessories also includes:

PSB-08 PROM System Board	119.95
regulated + 12 volts	124.95
B-08 2708 EPROM Programmer	99.95
regulated + 12 volts	104.95
U2708 EPROM Software	29.95
Casette tape	9.95
UIO Universal I/O Board	24.95
X-50 Extender Board	29.95
X-30 Extender Board	22.95

Visa and Master Charge Accepted

P.O. BOX 1110 DEL MAR, CA. 92014 714-756-2687

M31

A Name To Remember...

Wasatch

SEMICONDUCTOR PRODUCTS

for

44 PIN 5 VOLT
4.5" x 6.5" CARDS

RAM, PROM, EPROM, PARALLEL I/O, SERIAL I/O, A/D, D/A, CPU

Watch For New Product Releases

W13

25 SOUTH 300 EAST · SUITE 215 · SALT LAKE CITY, UTAH 84111 · 714/752-1374

Strings and Things

BASIC conversion techniques

Richard Roth
TSA Software
5 N. Salem Road
Ridgefield CT 06877

You have advanced far enough in programming to use character strings; yet, when you try to run a program using character strings from a book or article, you find half of them don't make any sense. If so, or if you are interested in handling characters in general, this article is for you.

A character string, basically a one-dimensional array or vector of characters, is a sequence of characters one after another. What distinguishes it from a vector of numbers is that it is used as a whole, rather than a character at a time.

In a game, the program may ask for someone's name, but it doesn't care that JOHN is a J followed by an O, H, N. The individual letters are considered a unit. In contrast, a mailing-list program that prints a list by last names scans MARYƀƀJ.ƀ-JONES to find the last word. It does this by scanning the characters until it finds a sequence of characters followed only by blanks. A space (represented by ƀ or blank) breaks the sequence of characters that comprise a word. We call such a break character a *delimiter.* Commas and periods also break the sequences of words into smaller units—phrases and sentences. A smaller unit of a character string is called a *substring.* Another special feature of character strings is length; a unit called NAME can vary from ED to STASTICOVICH. Usually, we fix a maximum length, but often we want to know the current length.

The problem arises when you want to use strings in BASIC, originally intended to work with numbers. Of course, a letter can be represented by a number, such as A = 1, B = 2 . . . or by the ASCII character set. In ASCII, digits (0-9), letters and special characters (such as Bell or Return) are all represented by a single integer from 0 through 127 (funny—it just fits in one byte!). In working with such simple numbers, BASIC wastes space because it is prepared for many digits of precision and doesn't know how simple a number is. Dealing with varying length and the string as a unit requires some built-in features. In the scientific language FORTRAN, the programmer must have a whole set of special subroutines to deal with strings.

When BASIC was first developed by Dartmouth's Kemeny and Kurtz, the only strings allowed were literals in print statements for title and labels such as: 100 PRINT "X = ",X. In early versions of BASIC, such as GE-635 Mark I Timesharing, extensions were added to allow the storage of strings, which were handled like single numbers. However, no advanced capability was available. A string array was specified by giving it a two-letter name. All one could do was print the

```
DIM AA(3)
READ AA(1), AA(2), AA(3)
DATA 'SALLY', 'JOE', 'SPOT'
PRINT 'A GIRL IS', AA(1)
PRINT 'A BOY IS', AA(2)
PRINT 'A DOG IS', AA(3)
```

Example 1.

HP BASIC	DEC BASIC
HP	DEC
Data General (DG)	Mits/Microsoft
North Star	BASIC-E
Computer Science Corp.	Tymshare
	Micro-polius (??)

Table 1.

HP	DEC
100 DIM N$(30),L$(10)	100 REM
110 S≠0 (state is beginning)	110 S=0
120 C=1 (character 1)	120 C=1
130 N$="SALLYƀƀJ.ƀJONES"	130 N$="SALLYƀƀJ.ƀJONES"
140 IF C>LEN(N$) GO TO 200	140 IF C>LEN(N$) GO TO 200
150 IF N$(C,C)="ƀ"THEN S=C	150 IF MID$(N$,C,1)="ƀ"THEN S=C
160 C=C+1	160 C=C+1
170 GO TO 140	170 GO TO 140
200 REM Now S=Char of last space	200 REM
210 L$=N$(S+1)	210 L$=RIGHT$(L$,S+1)

Example 2.

whole string. (See Example 1.)

This extension was short lived, but it set the stage for what we now have. The idea of uniquely specifying a string name became more prevalent, and '$' was finally accepted as the last character of a string name. But we are still plagued by the questions: How does one specify a single character of a string; and how does one specify a matrix of strings? Two primary approaches developed—one by Digital Equipment Corporation (DEC) and the other by Hewlett-Packard (HP) in the HP-200 series.

DEC emphasized many strings grouped as a matrix; and so A$(1) became the first element in a string matrix. HP emphasized each character in the string; and so A$(1) became the first character of string A$. These approaches led to a major difference in string handling. DEC BASIC requires a special way of getting a single character of a string, while HP BASIC must handle a string array specially. Table 1 shows a summary of the different BASICs.

From now on I will refer to the two schemes simply as HP or DEC, even though most schemes I will be referring to have not been written by either company.

What's It All Mean?

The issue involves how one deals with strings. For simple strings, such as printing the name of a single game-player, there is (almost) no difference (see Table 2).

Since HP BASIC uses the subscript notations to refer to substrings, DIM specifies the length of the string. DEC BASIC uses DIM to indicate how many strings in a string matrix; if no DIM occurs, it is just a single string (also called a scalar string, as opposed to a matrix).

DEC BASIC has no way of specifying maximum string length. They allow a maximum limit, usually 255 characters, set by the BASIC designer. HP BASIC tends to allow length limited only by memory size. Since HP BASIC knows how big a string can get, it can reserve a fixed space. DEC BASIC must constantly shift the strings around as lengths change. In this respect, HP BASIC enjoys a speed advantage.

Character Manipulations

There are two levels of manipulations—character and string. Each scheme of BASIC has its own home ground: character for HP and string for DEC. Getting at a single character is required for many functions. An early example suggested extracting a last name to alphabetize a mailing list. Example 2 shows this in both schemes.

To get at the fifth character of a string, HP BASIC uses only N$(5,5), while DEC BASIC uses MID$(N$,5,1). In our example, the program considers one character at a time from the string N$ with the name, until it gets to the end. Each time it sees a blank, it saves the character number in S. With no trailing blanks, the program, when it reaches line 200, S will point to the last blank. So the last name is S+1 through the end. (For simplicity, I am assuming a statement can follow an IF-THEN statement, as in most current BASICs.)

Table 3 shows how to get at a specific character. By HP rules the first subscript is the starting character, the second is ending character: if A$ = "ABCD", then A$ (2,3) = "BC". If no second subscript exists, then the rest of the string is used: if A$ = "ABCDEF" then A$(3)="CDEF". DEC uses functions MID$, RIGHT$, LEFT$ (as shown in the table) for the string A$="ABCDEF".

Single-character string functions are shown in Table 4.

String Manipulations

While DEC BASIC has awkward functions when dealing with substrings, HP BASIC has a far greater problem when many strings must be manipulated. It has no way to handle a group of strings of variable length. In the HP-3000 BASIC this was remedied in an elegant manner—especially true to BASIC syntax; unfortunately, only HP-3000 and Computer Science have implemented this

HP	DEC
DIM A$(30)	
INPUT A$	INPUT A$
PRINT "HELLO", A$	PRINT "HELLO", A$
RUN	RUN
?SAM (CR)	?SAM (CR)
HELLO SAM	HELLO SAM

Table 2.

DEC	HP
MID$(A$, starting char, length)	
MID$(A$,2,2)="BC"	A$(2,3)
LEFT$(A$, length)	
LEFT$(A$,3)="ABC"	A$(1,3)
RIGHT$(A$, starting char)	
RIGHT$(A$,3)="CDEF"	A$(3)

(Note: The parameter for RIGHT$ is starting character in DEC BASIC-plus, but length from the right in Mits BASIC and BASIC-E.)

Table 3.

Function		HP	DEC
Length		LEN(A$)	LEN(A$)
Substring	- 1 char at I	A$(I,I)	MID$(A$,I,1)
	- N chars at I	A$(I,I+N−1)	MID$(A$,I,N)
	- Char I to J	A$(I,J)	MID$(A$,I,J−I+1)
	- Char I to end	A$(I)	RIGHT$(A$,I)
	- Char 1 to Char I	A$(1,I)	LEFT$(A$,I)

Table 4.

in their BASICs. They have added:

```
DIM S(3)$(5)
A$=S(1)$(3,4)
```

The first subscript is a matrix; the second, a substring. Similarly, the first DIM value is matrix size; the second is maximum length.

Most people with HP BASIC can handle (with difficulty) a form of string matrix. Imagine the string V$, length 100, to be made of ten substrings, each ten characters long. The key is to fill out each string to a full ten characters; otherwise, the larger string will have holes. Creating one of those holes by putting in a shorter string will chop off the rest of the larger string. This also makes all the pseudomatrix elements a fixed length, which is annoying but better than no string matrix. For example, the fifth string in string matrix V$ is extracted by using: S5$ = V$(4*L + 1,5*L) (this is for a matrix starting at element 1). Two simple user functions will ease this calculation (see Example 3).

Concatenation

The second major function in string manipulation is concatenation, i.e., combining two strings to make one. For example, "HEL" + "LO" = "HELLO" (using DEC concatenation operator). HP has no common, direct way of doing this. Both + and , are allowed in some HP BASICs as concatenation operators. If no operator exists, HP BASIC allows a rather strange use of the subscript/substring to do this (see Table 5). At the L1 and L2 calculations, X$ is kept at full length and need not be refilled.

When using HP form strings for pseudostring matrices or concatenation, one must be very careful to fill out each string assignment where the subscript/substring is on the left side, i.e., S$(1,4) = A$. An improper assignment may chop off the end of the string on the left. This varies between HP-style BASICs, for example:

```
A$(5,9)=B$   where LEN(B$)<4
A$(5)=B$
```

In both cases, the length of A$ might become 5 + LEN(B$). (Data General had this problem before Release 3 RDOS BASIC, whereas, North Star Release 2 does not have the problem.)

Commands, Special Characters, Numbers and Input/Data

There are several less important differences that relate to assorted areas that vary between both schemes, all versions. Commands vary from BASIC to BASIC, for example, NEW or SCR (scratch), which is used to clear out an old program.

Getting special characters into and out of strings requires special care. Normally, a bell, for example, cannot be entered into a string. Some BASICs allow the code to be typed in a quoted literal. This can cause a problem because a listing will not show the character or, even worse, it will do the function (for example, turn on the paper-tape punch). One scheme by DG allows a special form in literal <#> in which the number is the internal form of the special code. For example, <7> is an ASCII BEL Code. The more common version allows a function, usually CHR$, that converts the numeric value to a string of the same character (BELL Code = CHR$(7)). The reverse function is ASC for ASCII value, where ASC("A") = 65 (the value of the letter A in the ASCII code). (Some BASICs use an ASCII null (true 0 byte) to indicate the end of a string. So A$(10) = CHR$(0) will chop off the string at 9 characters—if your BASIC does this.)

A similar conversion from internal to character string form is often available for numbers, too. NUM$(A) = "0.0" if A = 0 or VAL(A$) = 0 if A$ = "0.0". If your BASIC does not have a formatted print, these are useful in doing special output or input formatting. Read *your* manual before trying these functions; they might not do what you would expect. Depending on the BASIC, the following sequence could give a lot of trouble.

```
10 A=10
20 A$=NUM$(A)
30 F$="FILE"+A$
40 OPEN FILE F$
```

Some BASICs format a "NUM$" call exactly like output and put a space before the numeric string. For example, F$ = "FILEb10"—not "FILE10" Some special functions allow any string, expression or literal, while others must be a simple variable. (The difference between internal form of a number and ASCII byte or a character string can be confusing for the novice. 10 is not the same as "10" and if you are not sure why, find someone who knows. For example, a BEL code is an ASCII 7, not 7.0 or "7"—the difference depends on the function required.) Because it is not clear which is the "obvious way," both exist (see Example 4). DEC style says when the IF condition is true execute the rest of the statement; if it is false, continue on the next line.

```
DEF FNL(X)=(X-1)*L+1/DEF FNH(X)=(X*L)
A$(FNL(X),FNH(X)) references element X where:
X=subscript, L=length and A$ is pseudomatrix.
```

Example 3.

```
HP
100 IF A=B THEN PRINT "EQUALS"/GO TO 300
110 GO TO 400

DEC
100 IF A=B THEN PRINT "EQUALS": GO TO 300
110 GO TO 400
```

Example 4.

(from DEC BASIC-PLUS

```
A$="BCDEFAF"
INSTR(1,A$,"AF")=6       (6th char position)
INSTR(1,A$,"ABD")=0      (not found)
INSTR(6,A$,"F")=7        (start looking at 6th char)
```

Example 5.

```
DEC                           HP
10 A$="HEL"                   10 DIM X$(80)
20 B$="LO"                    20 X$="                " (80 blanks)
30 S$=A$+B$                   30 A$="HEL"
40 PRINT "STRING=",S$         40 B$="LO"
   Run                        50 L1=LEN(A$)
   STRING=HELLO               60 L2=LEN(B$)
                              70 X$(1,L1)=A$
                              80 X$(L1+1,L1+L2)=B$
                              90 S$=X$(1,L1+L2)
                              100 PRINT "STRING=",S$
                                 Run
                                 STRING=HELLO
```

Table 5.

5-190	Dimension and Functions
200-299	Read in names and Data Statements
300-499	Swap names, last name first
500-699	Bubble sort alphabetically
700-899	Print sorted list

Table 6.

Most HP BASICs only allow a line number after THEN. North Star says if true, execute the rest; if false, skip only the THEN clause, not the line. HP-style BASIC may or may not print "EQUALS," but it will always go to line 300; DEC style will only go to line 300 if "EQUALS" is printed; otherwise it will go to line 400.

Another feature of some BASICs is a string search, which locates a substring in a larger string (see Example 5).

Back to Reality

Let's condense all this discussion into one example which compares a list sort in HP-style and DEC-style BASIC. To add character functions we enter the list first name first and sort it first name last. Both are listed in Programs A and B and have approximately corresponding line numbers (see Table 6).

For the HP-like BASIC, we used North Star BASIC, which took 22 seconds from run to ready; the DEC-like BASIC was BASIC-E, which took 10 seconds (but it's a partial compiler). Neither time reflects a great sort but it works and illustrates our discussion here. (Fig. 1 is a run of the program.)

Peculiarities of the HP-like version are primarily related to the pseudomatrix required because the names functions FNL and FNH are used to calculate the start and end charac-

```
READY
LIST
100  REM WRITTEN IN NORTHSTAR BASIC (RELEASE 2)
110  READ N9
120  DIM N$(N9*30), F$(30),F1$(30),F2$(30),A$(30)
130  REM USE FUNCTIONS FOR PSEUDOMATRIX OF STRINGS
140  DEF FNL(X)=(X-1)*30+1 \ DEF FNH(X)=X*30
150  DEF FNA$(A$)
160  IF LEN(A$)>=30 THEN RETURN A$
170  A$=A$+"ƀ" \ GOTO 160 \ FNEND
200  REM IN NAMES
205  PRINT " **** NAMES ****" \ PRINT
210  N$="" \ REM CLEAR MATRIX
220  FOR I=1 TO N9
230  READ F$
235  PRINT F$
240  F$=F$+"$" \ REM MARK END OF NAME FOR REVERSE ROUTINE
250  F$=FNA$(F$)  \ REM FILL NAME TO 30 CHARS
260  N$=N$+F$
270  NEXT I
280  REM DATA
282  DATA 10
284  DATA "SALLY JONES", "SAM SMITH", "JOE SMITH", "TIM CAMBELL", "ED HILL"
286  DATA "STEVE MOODY", "ROGER HEAD", "SHIRLEY JONES", "ISSAC DEAR", "RICH KING"
300  REM RE-ORDER LAST NAME FIRST
310  FOR N1=1 TO N9
320  F$=N$(FNL(N1),FNH(N1))
330  C=1
335  REM LOOP UNTIL END MARK FOUND
340  IF F$(C,C)="$" THEN 380
350  IF F$(C,C)="ƀ" THEN S=C
360  C=C+1
370  GOTO 340
380  REM REVERSE FIRST & LAST NAMES
390  F1$=F$(1,S-1)           \ REM FIRST NAME
400  F2$=F$(S+1,C-1)         \ REM LAST NAME
410  F$=F2$+","+F1$
415  REM PUT BACK IN MATRIX (NOTE FULL 30CHARS SO NO LEFT-OVERS)
420  N$(FNL(N1), FNH(N1))=FNA$(F$)
430  NEXT N1
500  REM BUBBLE SORT, LOOP UNTIL NO SWAP ON A PASS
510  F=0
520  FOR I=2 TO N9
530  IF N$(FNL(I),FNH(I))>=N$(FNL(I-1),FNH(I-1)) THEN 590
540  REM SWAP
550  F=1      / REM REMEMBER A SWAP WAS DONE
560  F$=N$(FNL(I),FNH(I))
570  N$(FNL(I),FNH(I))=N$(FNL(I-1),FNH(I-1))
580  N$(FNL(I-1),FNH(I-1))=F$
590  NEXT I
600  IF F>0 THEN 510      \ REM KEEP TRYING TILL NO SWAPS
800  REM PRINT SORTED LIST
805  PRINT \ PRINT \ PRINT " **** SORTED NAMES ****" \ PRINT \ PRINT
810  FOR I=1 TO N9
820  F$=N$(FNL(I),FNH(I))
830  PRINT F$
840  NEXT I
850  END
READY
```

Program A. Mailing list (HP style).

```
MAILING. BAS WRITTEN IN BASIC-E (11/6/77)
5    REM WRITTEN IN BASIC-E
7    REM GET NUMBER OF NAMES
10   READ N9
15   DIM N$(N9)
200  REM READ IN NAMES
205  PRINT " **** NAMES ****"
210  FOR I=1 TO N9
220  READ N$(I)
225  PRINT N$(I)
230  NEXT I
240  DATA 10
250  DATA SALLY JONES, SAM SMITH, JOE SMITH, TIM CAMBELL, ED HILL
260  DATA STEVE MOODY, ROGER HEAD, SHIRLEY JONES, ISSAC DEAR, RICH KING
300  REM RE-ORDER LAST NAME FIRST
310  FOR N1=1 TO N9
320      C=1 : F$=N$(N1) : L=LEN(F$)
325      REM LOOP UNTIL LAST CHAR AND MARK LAST BLANK
330      IF C>L THEN 365
340          IF MID$(F$,C1)="ƀ" THEN S=C
350          C=C+1
360          GOTO 330
365      REM ACTUALLY SHUFFLE NAMES
370      F1$=LEFT$(F$,S-1)        : REM FIRST NAME
379          REM NOTE RIGHT$(NAME,LENGTH)
380      F2$=RIGHT$(F$,L-S)       : REM LAST NAME
390      F$=F2$+","+F1$
392          REM FILL OUT LENGTH SINCE 3 CHAR STR<4 CHAR STR
395      N$(N1)=F$+LEFT$("                              ",30-LEN(F$))
400  NEXT N1
500  REM DO SIMPLE BUBBLE SORT
510  F=0    : REM LOOP UNTIL NO SWAPS ON A PASS
520  FOR I=2 TO N9
530      IF N$(I)>N$(I-1) THEN 590
540          REM SWAP
550          F=1     : REM REMEMBER SWAP
560          F$=N$(I)
570          N$(I)=N$(I-1)
580          N$(I-1)=F$
590  NEXT I
600  IF F>0 THEN GOTO 500      : REM TEST FOR DONE
800  REM PRINT SORTED LIST
810  PRINT : PRINT : PRINT " **** SORTED LIST ****"
820  FOR I=1 TO N9
830  PRINT N$(I)
840  NEXT I
```

Program B. Mailing list (DEC style).

ters of a name element of 30 characters in the pseudomatrix N$ of names. FNA$ is used to fill a name out to 30 characters. Since a pseudomatrix element must be a fixed length, $ is used at the end of a name on initial entry so the first name/last name swap tells where the name ends.

The DEC-style version looks much nicer, primarily because it accepts a tab character while being typed in and thus is easier to format (called pretty-print). It is wise to do this if you can since it makes the reading of the program easier.

Line 380 uses the RIGHT$ function. This particular BASIC has the second parameter as the length, so RIGHT$ returns the right n-most characters (i.e., RIGHT$("ABCDEF",3) = "DEF"). Yet a true DEC-written BASIC will return from the nth character to the end (i.e., RIGHT$("ABCDEF",3) = "CDEF"). Line 395 illustrates one of the nice things about a DEC-like BASIC—string elements of variable length. This particular BASIC says a long string of As is greater than a short string of Bs, i.e., AAA>BB. Well, to each his own. (Note: This is specific to this BASIC (BASIC-E), not to all DEC-like BASICs.)

Summary

We have looked at two different ways of using strings in BASIC, both are common enough to have a following, but the most useful one is the one on your computer. Which is better? It's not for me to know; however, I have used both long enough to know that strings make a program really fun to use—even if it's a business program. That is because we talk in strings, not numbers. Like other computer users, I have braved strings in FORTRAN (which has no strings) and thrilled to a real string language like SNOBOL (running on a 360/65 in 250K). You use what you have! And hope someone's coming along with something better. Until then, keep on coding! ■

References

1. *Data General Extended BASIC User's Manual,* Rev 6, Feb. 1975.
2. *DEC PDP-11 BASIC-PLUS Language Manual,* July 1975.
3. *Altair BASIC Reference Manual,* 1975.
4. *Tymshare BASIC Tycom-X Manual,* March 1973.
5. *CTSTS BASIC Reference* (INFONET, Computer Sciences Corp.), May 1974.
6. *Timeshare BASIC/2000 Level F Reference Manual* (Hewlett-Packard), Feb. 1975.
7. *North Star BASIC Version 6 Manual,* Feb. 1977.
8. *Personal Notes from GE-635 Mark I Timesharing,* Oct. 1968.


```
     ****  NAMES  ****
SALLY JONES
SAM SMITH
JOE SMITH
TIM CAMBELL
ED HILL
STEVE MOODY
ROGER HEAD
SHIRLEY JONES
ISSAC DEAR
RICH KING

     ****  SORTED NAMES  ****
CAMBELL, TIM
DEAR, ISSAC
HEAD, ROGER
HILL, ED
JONES, SALLY
JONES, SHIRLEY
KING, RICH
MOODY, STEVE
SMITH, JOE
SMITH, SAM
READY
```

Fig. 1. List sort.

Thomas E. Doyle
5222 Big Bow Road
Madison WI 53711

5 Minutes or 5 Hours?

sorting techniques compared

Time Required to Sort N Items (seconds)

N	*Ripple*	*Modified*	*Bubble*	*S-M*
50	61	43	33	9
100	245	173	130	21
150	552	390	290	36
300	-	-	1224	85

Number of Swaps of Entries

N	*Ripple*	*Modified*	*Bubble*	*S-M*
50	1225	1225	1225	105
100	4950	4950	4950	260
150	11175	11175	11175	425
300	-	-	44850	1000

Number of Entry Comparisons

N	*Ripple*	*Modified*	*Bubble*	*S-M*
50	2450	1225	1225	263
100	9900	4950	4950	668
150	22350	11175	11175	1187
300	-	-	44850	2812

Table 1.

In an attempt to help justify the purchase of a floppy-disk system, I decided to put the computer to some practical use. It seems that not everyone considers piloting the Enterprise and destroying Klingons as a useful function worthy of another kilobuck investment. Using the system to keep track of household expenses seemed to be a good place to start. The Do-All program by Randy Miller (*Kilobaud*, August 1977) provided an ideal program.

After the program was loaded, a list of about a hundred items was entered for my demonstration of the practical advantages of a home computer. Everyone gathered for the show, and the program was run. A command was given to sort the list of data alphabetically. Everyone stared at the printer waiting for the output from this electronic marvel. Nothing happened.

Taking advantage of the pause and the presence of a captive audience, I discussed the advantages of adding a disk to the wonderful computer. At the end of my rather lengthy discussion there was still nothing on the printer. As time wore on, I began to consider the possibilities: hardware problems, software problems or simply another example of Murphy's Law. I felt there must be something wrong. After all, the Enterprise could move across the entire galaxy in only seconds, so alphabetizing this list could not take that long. Trying to remain cool, I suggested that we leave the computer and come back when it was done.

Much to my suprise, thirty minutes later the sorting was

```
5 REM --- RIPPLE SORT ---
6 REM --- SET UP ARRAY ---
10 N=150
20 DIM D(N)
30 J=N
40 FOR I=1 TO N
50 D(I)=J
60 J=J-1
70 NEXT
80 PRINT "*"
90 REM --- START OF SORT ---
100 M=N
105 C=0
110 FOR I=1 TO M-1
120 CM=CM+1
130 IF D(I)<=D(I+1) THEN 160
135 SW=SW+1
140 T=D(I):D(I)=D(I+1):D(I+1)=T
150 C=1
160 NEXT I
170 IF C=1 THEN 105
300 REM --- PRINT RESULTS ---
310 PRINT "SWITCHES =";SW
320 PRINT "COMPARISONS =";CM
330 PRINT "SIZE -";N
OK
```

Program A.

```
5 REM --- MODIFIED RIPPLE SORT ---
6 REM --- SET UP ARRAY ---
10 N=150
20 DIM D(N)
30 J=N
40 FOR I=1 TO N
50 D(I)=J
60 J=J-1
70 NEXT
80 PRINT "*"
90 REM --- START OF SORT ---
100 M=N
110 C=0
112 M=M-1
115 IF M=0 THEN 300
120 FOR I=1 TO M
125 CM=CM+1
130 IF D(I)<=D(I+1) THEN 160
135 SW=SW+1
140 T=D(I):D(I)=D(I+1):D(I+1)=T
150 C=1
160 NEXT I
170 IF C=1 THEN 110
300 REM --- PRINT RESULTS ---
310 PRINT "SWITCHES =";SW
320 PRINT "COMPARISONS =";CM
330 PRINT "SIZE -";N
OK
```

Program B.

```
5 REM --- BUBBLE SORT ---
6 REM --- SET UP ARRAY ---
10 N=150
20 DIM D(N)
30 J=N
40 FOR I=1 TO N
50 D(I)=J
60 J=J-1
70 NEXT
80 PRINT "*"
90 REM --- START OF SORT ---
100 M=N
110 FOR I=1 TO M-1
120 FOR J=I+1 TO M
125 CM=CM+1
130 IF D(I)<=D(J) THEN 170
135 SW=SW+1
140 T=D(I):D(I)=D(J):D(J)=T
170 NEXT J
180 NEXT I
300 REM --- PRINT RESULTS ---
310 PRINT "SWITCHES =";SW
320 PRINT "COMPARISONS =";CM
330 PRINT "SIZE -";N
OK
```

Program C.

complete. The printout revealed that the list had been sorted exactly as requested. What could have caused the delay? Perhaps my 8080 was slow. The benchmark programs in the basic timing comparisons article (*Kilobaud*, June 1977) were run and revealed that my computer ran a little faster than the one used for the article.

Since the program ran properly and the computer was up to speed, the solution to the problem must be in the sorting technique used in the program. An article on sorting routines by Andrew J. Rerko (*Kilobaud*, April 1977) was consulted and some test programs (Programs A, B and C) were run using the Ripple, Modified Ripple and Bubble routines described in the article.

The test programs consisted of setting up an array of N numbers in reverse order and using each of the sorting routines to sort them. The program execution times as well as number of comparisons and the number of element switches were recorded. The results are shown in Table 1. The results of this test revealed two things: The bubble sort was a little faster than the others, and sorting takes a lot of time. Sorting a simple table of 100 numbers took almost three minutes. No wonder the Do-All program took so long.

None of the common sorting methods described in Mr. Rerko's article would speed up a sorting program significantly. The solution to the problem, if any, would lie in an uncommon sorting routine. An article by John P. Grillo (*Creative Computing*, November 1976) discusses a technique called the Shell-Metzner Sort. This method offered significant speed advantages when sorting large amounts of data. A flowchart of the Shell-Metzner Sort is shown in Fig. 1. The article stated that a projected sort of 10,000,000 items would take 93 years using a bubble sort. Using the S-M technique, sorting the same data would require only 2.5 days. But would it help when sorting small amounts of data?

The benchmark sorting program was run using the S-M method and is shown in Program D. When sorting 150 items, the S-M sort was over eight times faster than the bubble sort and over 15 times faster than a ripple sort. The bubble sort required over 20 minutes to

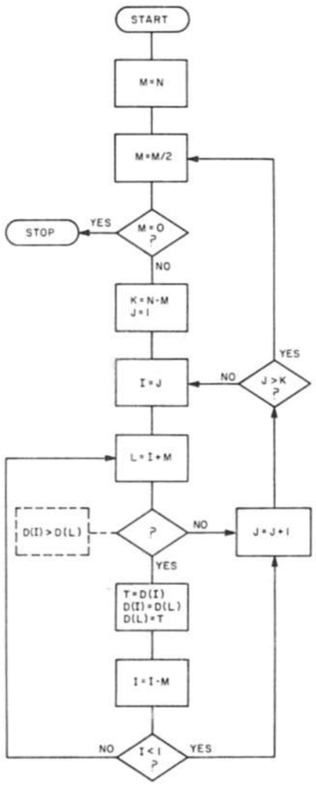


Fig. 1. Shell-Metzner Sort.

sort 300 items. The S-M method required only 85 seconds to sort the same list. The speed advantage of the S-M sort increases dramatically with the size of the list, but it seemed to speed sorts of even small lists.

The next step was to incorporate the S-M sort technique into the Do-All program and try it out. A random list of 100 entries was prepared and sorted by the standard program. Almost 45 minutes were required to sort this list. The Do-All program was then modified to use the S-M sort. Sorting the same list of 100 entries now required less than nine minutes. To modify the Do-All program, remove lines 4050-4115, 4150-4280, 9220-9340 and replace with the new lines shown in Progam E.

The only disadvantage I have found with the S-M technique so far is that it does require slightly more code, and it uses five index variables rather than only one or two as other sorting methods. Following the example benchmark program, it should be possible to use the S-M technique in other sorting programs.

Notes on Programs

All programs were run on an 8080 system with a 2 MHz clock and zero wait states. Mits 8K BASIC (Version 3.2) was used. Variable CM was used to total the number of comparisons between table entries. The variable SW was used to total the number of switches between table entries.■

```
5 REM --- SHELL METZNER SORT ---
6 REM --- SET UP ARRAY ---
10 N=300
20 DIM D(N)
30 J=N
40 FOR I=1 TO N
50 D(I)=J
60 J=J-1
70 NEXT
80 PRINT "*"
90 REM --- START OF SORT ---
100 M=N
110 M=INT(M/2)
120 IF M=0 THEN 300
130 J=1 : K=N-M
140 I=J
150 L=I+M
155 CM=CM+1
160 IF D(I)<D(L) THEN 210
170 T=D(I): D(I)=D(L): D(L)=T
175 SW=SW+1
180 I=I-M
190 IF I<1 THEN 210
200 GOTO 150
210 J=J+1
220 IF J>K THEN 110
230 GOTO 140
300 REM --- PRINT RESULTS ---
310 PRINT "SWITCHES =":SW
320 PRINT "COMPARISONS =";CM
330 PRINT "SIZE -";N
OK
```

Program D.

```
LIST 4050

4050 M=P
4055 M=INT(M/2)
4060 IF M=0 THEN 1140
4065 J=1 : K=(P-1)-M
4070 I=J
4075 L=I+M
4080 IF N(T,I)<=N(T,L) THEN 4105
4085 GOSUB 9210
4090 I=I-M
4095 IF I<1 THEN 4105
4100 GOTO 4075
4105 J=J+1
4110 IF J>K THEN 4055
4115 GOTO 4070
BREAK
OK
LIST 4150

4150 M=P
4160 M=INT(M/2)
4170 IF M=0 THEN 1140
4180 J=1 : K=(P-1)-M
4190 I=J
4200 L=I+M
4210 IF A$(T,I)<=A$(T,L) THEN 4260
4220 GOSUB 9210
4230 I=I-M
4240 IF I<1 THEN 4260
4250 GOTO 4200
4260 J=J+1
4270 IF J>K THEN 4160
4280 GOTO 4190
BREAK
OK
LIST 9220

9220 X1=N(1,L)
9230 X2=N(2,L)
9240 B1$=A$(1,L)
9250 B2$=A$(2,L)
9260 FOR Z=1 TO 2
9270 N(Z,L)=N(Z,I)
9280 A$(Z,L)=A$(Z,I)
9290 NEXT
9300 N(1,I)=X1
9310 N(2,I)=X2
9320 A$(1,I)=B1$
9330 A$(2,I)=B2$
9340 RETURN
BREAK
OK
```

Program E.

Do-It-Yourself Time-sharing

it's easier than you think

Mike Kop
3060 Marshall Ave.
Cincinnati OH 45220

When I first learned to program I was taught how to sign on to a computer system using a teletypewriter, type in a program and obtain the results at the terminal almost immediately. Other users around me, each working on his own program, were using similar type terminals. It appeared that each user had the entire computer to himself! This amazed and perplexed me. How could a computer run all the terminals and keep track of everybody—all at the same time? I conceded that the system was too complex to analyze (or perhaps it was sheer magic). Eventually I began to understand what went on by fighting my way through books on operating systems. I hope that future computer users will be spared a similar experience.

Last year, I purchased an M6800 system from SWTP. After programming on it for a while, I decided to investigate the possibility of implementing time-sharing on my system. It turns out to be simpler than you might think.

In this article, I will attempt to explain exactly why one would want to set up time-sharing and how it is done (for an M6800 system). I'll also try to explain some other programming considerations.

What is Time-sharing?

Time-sharing is accomplished by switching rapidly between many users. That means each user is allowed, in turn, a short duration of central processing unit (CPU) or microprocessing unit (MPU) time. This is called a time slice. For example, if the time slice were 50 milliseconds, then each user would use the processor for 50 milliseconds. If the switching is fast enough, the computer operation from each user's point of view will appear continuous.

Why Time-sharing?

The computer in a large system may cost several million dollars. Obviously, buying one computer for each user is extremely impractical. Sharing the computer among many users is a more effective way to utilize the system.

Another reason for time-sharing is because a computer's input/output (I/O) devices are much slower than the processor. If a terminal is outputting characters at 30 cps, there is sufficient time between characters for other work. Thus, with time-sharing, literally two, three or more times as much work can be accomplished than by a single user.

Most of the reasons given for using time-sharing would also apply to a microcomputer system (perhaps on a smaller scale). One possible argument against its use in microprocessors would be that they're too slow. However, for programs that do a lot of input and output and use little processor time (most games and business-type programs fall into this category), I see no reason why time-sharing cannot be implemented.

Using Interrupts

Proper use of interrupts comes first in implementing time-sharing. The ideas presented here are essentially the same, whether you have a small or large system.

An interrupt is basically a hardware mechanism that makes the microprocessor stop what it is doing and jump to another program (often known as a service routine). Sometimes it is possible to mask off an interrupt. If this happens, then the interrupt is ignored (or held pending until some later time).

Let's look briefly at the interrupt mechanisms on the SWTP system (which uses MIKBUG). There is a line marked IRQ (for interrupt request). If this line is temporarily gounded and the mask bit is a zero, an interrupt will occur. The system will then jump to the address contained in storage locations $A000 and $A001. One nice thing about the M6800 microprocessor is that when interrupted it stores everything (i.e., the condition code, B, A, X and program counter registers) on the stack. This means that little effort is required to remember where each program is when it was stopped. With other processors, you would typically have to store all registers away, which may take many instructions. One danger of this is that if another interrupt occurs before all registers are stored away, some register contents may be lost. The M6800 processor saves everything in one swoop.

Incidentally, you may, if desired, use the nonmaskable interrupt NMI instead of IRQ. The interrupt address would then be stored at locations $A006 and $A007. I prefer, however, to use an interrupt that is maskable.

Software

Program A actually implements time-sharing. The comments should aid you in understanding how the program works. It starts at address BEGIN. Also, some hardware must be set up so that an IRQ interrupt is generated at regular intervals (this is explained later). Each time an interrupt is generated, one program is stopped and the next one in line is started. For example, if program 1 is currently executing and we are time-sharing three programs, then four interrupts will result in program 2 being executed (1 then 2 then 3 then 1 then 2). With Program A, you may time-share up to 15 different programs.

The part of the program that actually does the time-sharing (the service routine) is statements 69 to 83. Statements 1 to 64 merely initialize various parameters. The initialization routine basically works thus—initially each program is assigned a stack pointer. The stack-pointer addresses differ by 16 bytes. That is, program 1 has a stack-pointer value of END + 16, program 2 has a value of END + 32, etc. These values are stored at addresses STACK1, STACK2, etc.

The initialization routine also clears the condition-code register and stores the starting address of each program at the appropriate position in each stack. When the RTI instruction is executed, the processor fetches all registers (program counter included in the fetch) from the stack and starts (or resumes) a program at the appropriate address.

The purpose of clearing the condition code in the stack for each program is that when the RTI instruction is executed, the interrupt mask bit will not become set (which would lock up the system). For example, if the stack pointer were at $0F00, we would clear address $0F01 and store the starting address at address $0F06. An RTI instruction would then load the condition-code register with $00 and the program counter with the number at address $0F06. Initially, we don't care what the other register contents are.

The service routine performs a very simple function. It stops the current program from executing and runs the next program in line; it accomplishes this by storing away the current stack pointer and loading the next one. When the RTI instruction is executed, we do not return exactly where we left off (that is, resume execution of the same program) as is normally done. Instead, we go to the next program. This occurs because the stack pointer has been changed.

You will also observe that in the service routine, I purposely store data where instructions are. This is a trick I use to make the service routine execute quickly, although in general this is not good practice. I do have another version of the service routine that does not do this; however, it is slightly longer.

For a simple demonstration of time-sharing, Program B may be used. This program assumes that you have a serial interface port (which uses an ACIA) at the correct baud rate at address $8008. You will also have to have a terminal plugged in at this address. We will call this terminal 2. Terminal 1 will be at the control interface. If you run the Program B starting

Program A. Time-share program.

STMT	ADDR	CODE		STATEMENT				
1	0E00				ORG		$0E00	
2	0E00			STRING	EQU		$E07E	
3	0E00			IN2HEX	EQU		$E055	
4	0E00			INHEX	EQU		$E0AA	
5	0E00			CR	EQU		$0D	
6	0E00			LF	EQU		$0A	
7	0E00			EOT	EQU		$04	
8	0E00	CE	0E92	BEGIN	LDX		#SERVCE	
9	0E03	FF	A000		STX		$A000	INITIALIZE INTERRUPT REQUEST POINTER
10	0E06	CE	0E61		LDX		#MES1	
11	0E09	BD	E07E		JSR		STRING	PRINT '#PROGRAMS = '
12	0E0C	BD	E0AA		JSR		INHEX	GET NUMBER OF PROGRAMS TO BE TIME SHARED
13	0E0F	B7	0EB2		STA	A	NUMBER	
14	0E12	16			TAB			
15	0E13	CE	0EB3		LDX		#STACK1	
16	0E16	FF	0E5D		STX		TEMP0	
17	0E19	CE	0EE1		LDX		#END + 16	X-REG NOW POINTS TO THE BEGINNING
18	0E1C	FF	0E5F		STX		TEMP	OF THE STACK AREA
19	0E1F	FE	0E5D	A1	LDX		TEMP0	LOAD ADDRESS OF STACK I
20	0E22	FF	0E2E		STX		ST0 + 1	
21	0E25	08			INX			
22	0E26	08			INX			
23	0E27	FF	0E5D		STX		TEMP0	STORE ADDRESS OF STACK I + 1
24	0E2A	FE	0E5F		LDX		TEMP	
25	0E2D	FF	FFFF	ST0	STX		$FFFF	INITIALIZE STACK I
26	0E30	6F	01		CLR		1,X	CLEAR CONDITION CODE REGISTER I
27	0E32	86	06		LDA	A	#6	
28	0E34	8D	22		BSR		ADD	
29	0E36	FF	0E4A		STX		ST + 1	THE X-REG NOW POINTS TO THE ADDRESS WHERE THE
30	0E39	86	0A		LDA	A	#10	STARTING ADDRESS OF PROGRAM I STARTS
31	0E3B	8D	1B		BSR		ADD	
32	0E3D	FF	0E5F		STX		TEMP	THE ADDRESS OF THE NEXT STACK WILL BE 16
33	0E40	CE	0E73		LDX		#MES2	BYTES AWAY FROM THE CURRENT STACK
34	0E43	BD	E07E		JSR		STRING	PRING 'START = '
35	0E46	BD	0E7C		JSR		INPUTX	INPUT STARTING ADDRESS
36	0E49	FF	FFFF	ST	STX		$FFFF	INITIALIZE PROGRAM COUNTER 1
37	0E4C	5A			DEC	B		
38	0E4D	26	D0		BNE		A1	

```
0E4F   86     01              LDA  A  #1          BEGIN RUNNING PROGRAM#1
0E51   B7     0EB1            STA  A  STATUS
0E54   BE     0EB3            LDS     STACK1
0E57   3B                     RTI
0E58   08           ADD       INX                 THIS SUBROUTINE INCREASES THE X-REG
0E59   4A                     DEC  A              BY THE VALUE IN THE A-REG
0E5A   26     FC              BNE     ADD
0E5C   39                     RTS
0E5D                TEMP0     RMB     2
0E5F                TEMP      RMB     2
0E61   0D0A         MES1      FCB     CR,LF
0E63   2350                   FCC     /#PROGRAMS(1-F)?/
       524F
       4752
       414D
       5328
       312D
       4629
       3F
0E72   04                     FCB     EOT
0E73   0D0A         MES2      FCB     CR,LF
0E75   5354                   FCC     /START=/
       4152
       543D
0E7B   04                     FCB     EOT
0E7C   36           INPUTX    PSH  A              SUBROUTINE TO INPUT THE X-REG
0E7D   37                     PSH  B
0E7E   BD     E055            JSR     IN2HEX
0E81   B7     0E90            STA  A  DATA
0E84   BD     E055            JSR     IN2HEX
0E87   B7     0E91            STA  A  DATA+1
0E8A   FE     0E90            LDX     DATA
0E8D   33                     PUL  B
0E8E   32                     PUL  A
0E8F   39                     RTS
0E90                DATA      RMB     2
                    *
                    *THE PURPOSE OF THIS SERVICE ROUTINE IS TO
                    *STOP PROGRAM I AND BEGIN RUNNING PROGRAM I+1
0E92   CE     0EB1  SERVICE   LDX     #STACK1-2
0E95   B6     0EB1            LDA  A  STATUS      DETERMINE THE PROGRAM CURRENTLY EXECUTING
0E98   48                     ASL  A
0E99   B7     0E9D            STA  A  ST1+1
0E9C   AF     00    ST1       STS     X           SAVE THE STACK POINTER AT THE APPROPRIATE
0E9E   47                     ASR  A              ADDRESS(STACK1,STACK2,. . .,STACKF)
0E9F   4C                     INC  A              BEGIN TO EXECUTE THE NEXT PROGRAM
0EA0   B1     0EB2            CMP  A  NUMBER      CHECK FOR WRAP AROUND
0EA3   2F     02              BLE     L3
0EA5   86     01              LDA  A  #1          IF WRAP AROUND EXISTS EXECUTE PROGRAM#1
0EA7   B7     0EB1  L3        STA  A  STATUS      INDICATE THAT THE NEXT PROGRAM IS EXECUTING
0EAA   48                     ASL  A
0EAB   B7     0EAF            STA  A  ST2+1
0EAE   AE     00    ST2       LDS     X           LOAD THE APPROPRIATE STACK POINTER
0EB0   3B                     RTI                 BEGIN ACTUAL EXECUTION
0EB1                STATUS    RMB     1           CURRENT PROGRAM IN EXECUTION(1 TO F)
0EB2                NUMBER    RMB     1           TOTAL NUMBER OR PROGRAMS TO BE TIME SHARED
0EB3                STACK1    RMB     2           STACK POINTER FOR PROGRAM#1
0EB5                STACK2    RMB     2           STACK POINTER FOR PROGRAM#2
0EB7                STACK3    RMB     2           . . . ETC . . .
0EB9                STACK4    RMB     2
0EBB                STACK5    RMB     2
0EBD                STACK6    RMB     2
0EBF                STACK7    RMB     2
0EC1                STACK8    RMB     2
0EC3                STACK9    RMB     2
0EC5                STACKA    RMB     2
0EC7                STACKB    RMB     2
0EC9                STACKC    RMB     2
0ECB                STACKD    RMB     2
0ECD                STACKE    RMB     2
0ECF                STACKF    RMB     2
                    *
0ED1                END       EQU     *
                    *
A048                          ORG     $A048
A048   0E00                   FDB     BEGIN
```

SYMBOL	VALUE	DEFN	REFERENCES	
STRING	E07E	2	11	34

at address $0000, a series of zeros should be printed out on terminal 2. Starting at address $0008 will result in a printout of all ones.

We will now time-share both parts of this program. For this part, first press the reset button. This will set the mask bit to a one. Now set the interrupt rate to a very slow value, say once every ten seconds if possible. (We'll discuss the hardware to accomplish this in a moment.) Now run Program A, starting at address BEGIN ($0E00). You will then be required to type in the number of programs you want (this is a single hex number from 1 to F) to time-share, followed by their respective starting addresses. The data is entered as follows:

```
#PROGRAMS(1-F)?2
START=0000
START=0008
```

After having done the above, you should see the printout at terminal 2 alternate between strings of zeros and strings of ones. If you slowly increase the interrupt rate you will notice that the respective strings become shorter and shorter.

If you do not have a second terminal, you may unplug the terminal from the control interface in each of the above steps and plug it into the other port after having typed a G. Be very careful when doing this; you should avoid the practice in general.

Perhaps you have wondered why I used another I/O port and not MIKBUG directly. MIKBUG outputs a character by software, bit by bit. If you were to interrupt the output routine, the output bits would not appear at the proper time. That is, you cannot output part of a character now and the other part later. This problem does not occur with an ACIA because a character is output by a single store instruction.

Hardware

As stated previously, interrupts must be generated at regular intervals. An interrupt should be generated by a pulse that grounds the IRQ line for a very short duration before returning to a high state. This is because the IRQ line must return to its high state before the service routine has completed its job. If this is not done, then another interrupt will occur immediately after the service is completed, causing some programs to be skipped in execution. A pulse duration of 50 microseconds works quite well. An interrupt will not occur inside the service routine because the mask bit will be set at that time. If, however, you decide to use NMI instead, your pulse must be much narrower (e.g., 10 microseconds). Otherwise, the service routine may keep interrupting itself, which can lead to difficulties!

If you have a signal generator that can generate a pulse, so much the better. I also understand that SWTP now has available an interrupt timer board. In place of these alternatives, you may use the circuit shown in Fig. 1. There are no doubt other circuits that will work as well. Resistors R1 and

IN2HEX	E055	3	57	59	
INHEX	E0AA	4	12		
CR	000D	5	49	52	
LF	000A	6	49	52	
EOT	0004	7	51	54	
BEGIN	0E00	8	105		
A1	0E1F	19	38		
ST0	0E2D	25	20		
ST	0E49	36	29		
ADD	0E58	43	28	31	45
TEMP0	0E5D	47	16	19	23
TEMP	0E5F	48	18	24	32
MES1	0E61	49	10		
MES2	0E73	52	33		
INPUTX	0E7C	55	35		
DATA	0E90	65	58	60	61
SERVCE	0E92	69	8		
ST1	0E9C	73	72		
L3	0EA7	79	77		
ST2	0EAE	82	81		
STATUS	0EB1	84	40	70	79
NUMBER	0EB2	85	13	76	
STACK1	0EB3	86	15	41	69
STACK2	0EB5	87			
STACK3	0EB7	88			
STACK4	0EB9	89			
STACK5	0EBB	90			
STACK6	0EBD	91			
STACK7	0EBF	92			
STACK8	0EC1	93			
STACK9	0EC3	94			
STACKA	0EC5	95			
STACKB	0EC7	96			
STACKC	0EC9	97			
STACKD	0ECB	98			
STACKE	0ECD	99			
STACKF	0ECF	100			
END	0ED1	102	17		

STMT	ADDR	CODE		STATEMENT			
1	0000	8D	0E	PRGRM1	BSR		SETUP
2	0002	86	30	LOOP1	LDA	A	#'0
3	0004	8D	18		BSR		OUTPUT
4	0006	20	FA		BRA		LOOP1
5	0008	8D	06	PRGRM2	BSR		SETUP
6	000A	86	31	LOOP2	LDA	A	#'1
7	000C	8D	10		BSR		OUTPUT
8	000E	20	FA		BRA		LOOP2
9	0010	FE	001C	SETUP	LDX		ACIA
10	0013	86	13		LDA	A	#$13
11	0015	A7	00		STA	A	0,X
12	0017	86	11		LDA	A	#$11
13	0019	A7	00		STA	A	0,X
14	001B	39			RTS		
15	001C	8008		ACIA	FDB		$8008
16	001E	DE	1C	OUTPUT	LDX		ACIA
17	0020	C6	02	T1	LDA	B	#$02
18	0022	E4	00		AND	B	0,X
19	0024	27	FA		BEQ		T1
20	0026	A7	01		STA	A	1,X
21	0028	39			RTS		

SYMBOL	VALUE	DEFN	REFERENCES	
PRGM1	0000	1		
LOOP1	0002	2	4	
PRGRM2	0008	5		
LOOP2	000A	6	8	
SETUP	0010	9	1	5
ACIA	001C	15	9	16
OUTPUT	001E	16	3	7
T1	0020	17	19	

Program B. Test program.

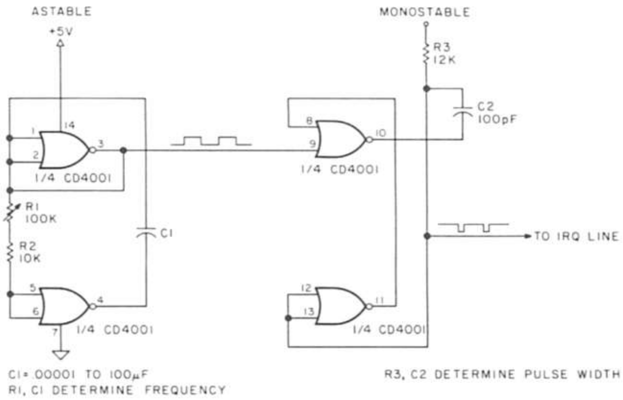


Fig. 1. Interrupt-oscillator circuit.

C1 may be changed to vary the interrupt rate.

The question of how often we generate an interrupt now arises. Suppose we were to generate an interrupt once every ten seconds. If each user were printing out data, the printing would be done in spurts. Another problem would be that a user might type in data while another program was being run, resulting in input being lost. If we increased the interrupt rate fast enough, the output would appear smooth and continuous. Also it would be impossible for a person to type so fast that some data might be lost. So, it would seem that the faster we generate interrupts, the better.

The problem, however, is that the service routine takes a fixed amount of time to perform its duties. As we increase the rate of interrupting, the percentage of time the microprocessor is in the service routine increases. It is possible to generate interrupts so fast that 99 percent of the time is spent in the service routine, meaning that only one percent of the processing time actually performs useful work. Therefore, we should try to choose an optimal interrupt rate. I find that 100 interrupts per second works well. You should experiment to determine what works best for you. You could also determine the optimal rate mathematically; this would require that you examine matters in more detail.

Programming Considerations

Suppose you are time-sharing two or more programs at the same time. If these programs are in different segments of memory, there are no problems. Often, however, it is desirable that programs be able to share the same subroutines; this is necessary for large programs.

For example, BASIC might take up approximately 8K bytes. If each of four users had his own copy of BASIC, we would need at least 32K! If all four users could use one copy of BASIC at the same time we would need only 8K, resulting in a tremendous saving in memory (of course, each user still needs his own area to store his program).

But wait a minute! You cannot take any subroutine and expect it to work on a time-shared basis. As a matter of fact, most subroutines would not work at all. A subroutine that is reentrant is needed. A reentrant subroutine is defined as one that may be employed by many users at the same time (i.e., on a time-shared BASIC). Let's go over some examples of reentrant and non-reentrant subroutines.

Let's say we wanted to write a subroutine that would add the contents of the A register to that of the B register and store the result in the B register. It is also desired that the A register not be modified when we return from this subroutine. The subroutine in Program C will accomplish this for a single user and will prove to be non-reentrant.

Suppose two users call this routine at about the same time, and the values of the A register for both users are $01 and $02, respectively, upon entry into the subroutine. User 1 enters the subroutine and executes the first three instructions before an interrupt occurs. Location TEMP will then contain a value of $01.

Let us now assume that after the interrupt, program 2 enters the subroutine and is interrupted after three instructions have been executed. Location TEMP now has a value of $02. After the interrupt, user 1 will resume execution and execute statement 4, a load instruction. The A register will now contain a value of $02. We will then return from the subroutine.

You will immediately notice that from user 1's point of view, the value of the A register has been changed from $01 to $02 upon leaving the subroutine. This was not intended. So, we have here an example of a subroutine that works for one user, but falls apart for two.

Now, let us write the same subroutine in a different way, as shown in Program D. This subroutine turns out to be reentrant. We'll assume the same sequence of events as in the previous example. User 1 will save $01 by pushing it onto its own stack. When user 2 enters the subroutine, it saves $02 on its own stack. The crucial point here is that each program has its own stack. Consequently, $01 and $02 are stored in different locations. When each program executes the PUL A instruction, it does so with respect to its own stack. This means that the proper values are restored. Two or more users can therefore use this subroutine at the same time!

Another example of reentrant programming can be found in the Motorola M6800 Programming Manual. For example, on pages 10-12 a reentrant 16-bit multiplication subroutine is depicted. The key technique here is that everything is first pushed onto the stack. The TSX (Transfer Stack Pointer to Index) is then executed. All instructions that follow are executed in the indexed mode. This is equivalent to the work area being in the stack. Nowhere in the program is there a label designating a storage location.

```
STMT  ADDR    CODE          STATEMENT
  1   0000    B7    0009    ADD    STA   A   TEMP    SAVE A-REGISTER
  2   0003    1B                   ABA
  3   0004    16                   TAB
  4   0005    B6    0009           LDA   A   TEMP    RESTORE A-REGISTER
  5   0008    39                   RTS
  6   0009                  TEMP   RMB       1

SYMBOL   VALUE   DEFN   REFERENCES
ADD      0000      1
TEMP     0009      6      1    4
```

Program C. A non-reentrant subroutine.

STMT	ADDR	CODE	STATEMENT			
1	0000	36	ADD	PSH	A	SAVE A-REGISTER
2	0001	1B		ABA		
3	0002	16		TAB		
4	0003	32		PUL	A	RESTORE A-REGISTER
5	0004	39		RTS		

SYMBOL	VALUE	DEFN	REFERENCES
ADD	0000	1	

Program D. A reentrant subroutine.

In general, writing reentrant subroutines may be easy or difficult, depending on the type of instruction set available. For example, if the M6800 microprocessor had a PSH X instruction, the task of reentrant programming would be greatly simplified. Other processors have defects of their own. Perhaps in the future someone will design a stack-oriented microprocessor. Reentrant programming may then become a trivial task. Incidentally, stack processors have other advantages than the one given.

You must be careful, though, that the stack pointer does not change too much from its initial value. At the start of execution, the stack pointers of all programs initially differ by 16. This will change slightly throughout the course of execution. For example, if we were in program 1, an interrupt might occur after we had jumped to a subroutine. This would cause the stack pointer to differ by 2 from its initial value. If we nested subroutines too deeply, say 8 or 9, we could change the stack pointer so much that we'd wipe out the stack of another program! This problem can be solved, however, by initially separating the stack pointers by more than 16.

Since the time-sharing routine uses the stack pointer for its own bookkeeping, you must be careful what you do with the stack pointer. A common technique is to use the stack pointer to point to a list of numbers. This will not work if the stack pointer is pointing to, say, the middle of a list of numbers. It won't work because on interrupt, the registers that are stored in the stack will destroy some numbers in the list. Jumping to a subroutine or doing PSHES and PULLS modify the stack pointer but are not harmful because the stack pointer is changed in a way that won't change valid data in the stack.

Remarks

In this article, I have tried to point out some of the essential points that must be understood in order to implement time-sharing. I hope I've taken some of the mystery out of it.■

Cassette Recorder Disaster: Ground Loops

the problem, and a solution

Photo 1. Cable adapter made from standard parts.

Dave Waterman
834 Oak Lee Ln.
Alpine CA 92001

Dave Lien
8662 Dent Dr.
San Diego CA 92119

Photo 2. The ungrounding adapter is inserted between the computer and the recorder in the EAR or AUX lines.

The ordinary household cassette recorder was not designed with anything as exotic as digital data recording in mind. Computer experimenters pressed the recorder into this role. All things considered, the device works well. However, two problems immediately arise—low-level ground loops, which can badly degrade the system's reliability, and the lack of a convenient means of overriding the computer's control of the drive motor. We'll address the problems separately.

Why the Ho Hum

The standard cassette recorder was not designed to input audio (data or otherwise) via its AUX or MIC jack, and an instant later feed audio out through the EAR jack—with all jacks tied to a common external ground. Many recorders do not even have a common *internal* ground for these jacks and the REM motor control jack. Those that do usually have a relatively high-resistance ground. When this shaky ground system is tied to the computer's common ground by way of three separate shielded cables (DATA-in, DATA-out and REMOTE motor control), the ground loops created can completely destroy the reliability of the recording system.

A Way Around this Hummer

The standard way out of this ground-loop problem is to unplug either the DATA-in or DATA-out plug from the recorder, whichever is not in use. It usually works but is inconvenient, particularly for the half-way serious computer user who values his time. Fortunately, there are a couple of simple and inexpensive solutions (until more suitable recorders hit the market at the right price).

Photo 1 shows a simple cable adapter made from standard parts. It consists of a miniplug, minijack and a short (the shorter the better) piece of *unshielded* wire. This wire is soldered *only* to the "hot" (center) connectors of both plug and jack.

This ungrounding adapter is inserted between the computer and the recorder in the EAR line or the AUX line, as shown in Photo 2. Given the choice, it is better to use an unbroken shield to the AUX jack to assure a good-quality recording. A properly recorded tape can always be reloaded, but a bad tape cannot. Keep power supplies and other possible sources of interference away from this unshielded adapter. It works well.

The second ground-looping solution is a variation on the same theme, but it also solves the annoying problem of lack of convenient motor control. Two jacks, one miniature (to match

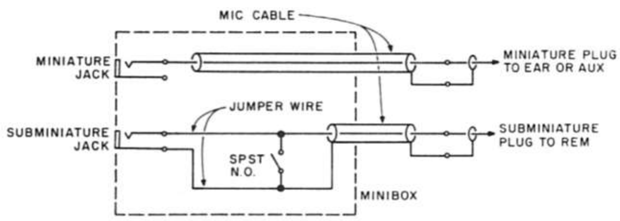


Fig. 1.

the EAR plug) and one subminiature (to match the REM motor plug), are mounted in a small plastic case. The one shown in Photo 3 was used to hold a burglar-alarm panic switch. A shielded cable is run from the EAR jack in the box to the EAR plug for the recorder. Note in Fig. 1 that the shielded part of the cable is not attached to break the ground loop. Another shielded cable is run from the REM jack to the REM plug for the recorder, but its ground integrity is maintained.

Similar switch boxes are equipped with an SPST normally closed switch. If this is the case with the one you select, replace the switch with a similar SPST switch with normally *open* contacts, as shown in Fig. 1. Unshielded jumper wires are then connected from the switch to the subminiature REM Jack-in-the-box (sorry about that!). Paralleling the REM line with the push-button switch allows us to turn on the motor.

We can always turn the recorder *off* with its normal STOP button. This arrangement allows us to turn the motor on for purposes of rewinding tape, advancing a cassette past the leader or going fast forward to find a certain spot on the tape.

Photo 4 shows this handy auxiliary control box installed with a Radio Shack TRS-80 computer system. It should work as well with any other.

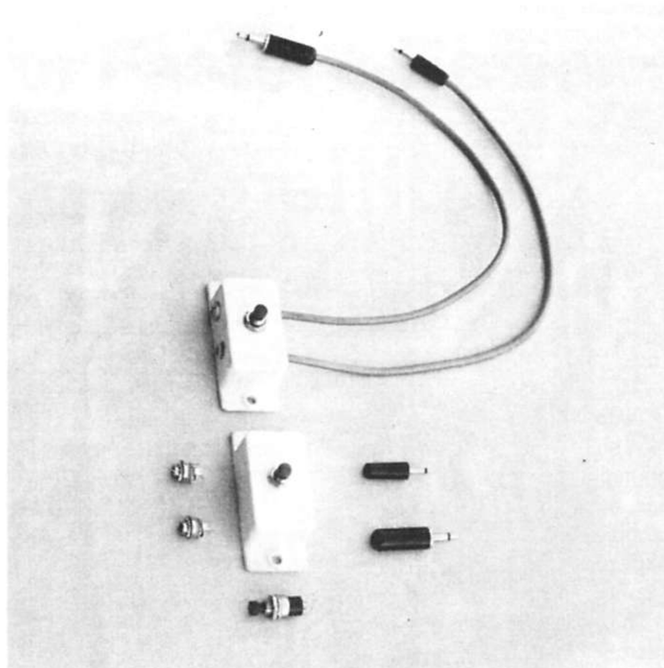


Photo 3. Small case with mounted jacks.

Success

Both of these solutions to nuisance problems work well, are inexpensive and require no special tools or skill. Give them a try, and see how much more you enjoy your computer. ■

Photo 4. The control box installed with a TRS-80.

Glen Charnock
864 Palomar Way
Oxnard CA 93030

A Different Search Technique

don't just try it—benchmark it

Good things *can* come in small packages. This programming trick is so simple it can easily retrofit to existing programs; yet, it can substantially reduce the time needed to search a table.

The traditional method of searching a table is shown in Fig. 1. First, a loop index is initialized. Then a loop is executed, comparing the table element with the search argument and incrementing the loop index until either a match is found or the table is exhausted. When the loop is exited, the loop index points either to the location of the matching table element or, if no match was found, to the last table element plus one.

The new method dimensions one extra place at the end of the table for a "dummy" value. To search the table, first move the search value into this dummy location at the end of the table; then initialize the loop index and begin looping through the table. This time, however, only search for a match and increment the loop index within the loop. You don't need to test for the end of the table . . . if you haven't found a match by then, you will on the last table entry because you've already moved the search argument into this last entry. Thus, you save one comparison for each table entry searched (see Fig. 2).

Depending on the language and the way the computer implements subscripts, this trick can save as much as half the the time needed for the search. That's pretty good for such a small change!

I learned this programming trick from the advertising brochure of Software Consulting Services of Allentown PA. Further details may be found in *The Art of Computer Programming*, Vol. 3, "Sorting and Searching," by Donald E. Knuth.■

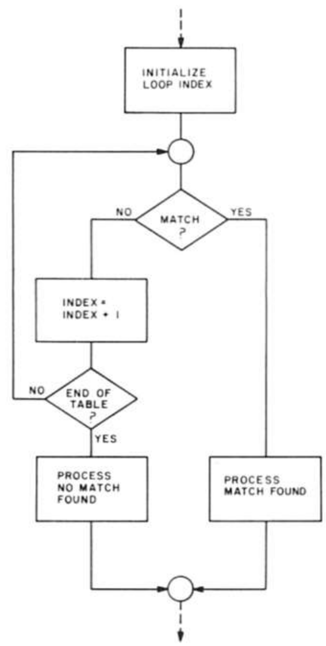


Fig. 1. Traditional table-searching method.

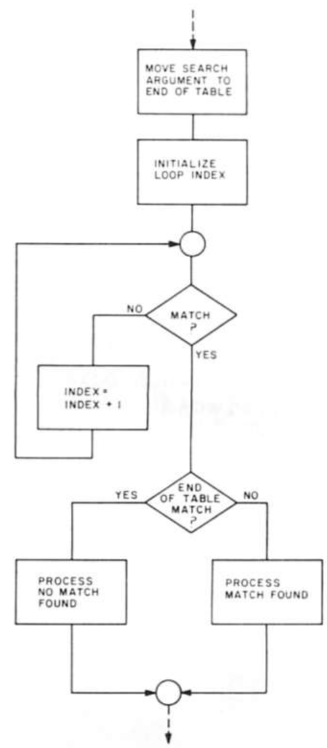


Fig. 2. A different search technique.

Kilobaud Classified

Kilobaud classified advertisements are intended for use by those individuals desiring to buy, sell, or trade used computer equipment or software. No commercial ads are accepted.
Two sizes of ads are available. The $5 box allows five lines of about 22 characters each, including spaces and punctuation. The $10 box provides ten lines of type – again, each line is about 22 characters. Minimize capital letters, as they use twice the space of small characters. Payment is required in advance with ad copy. We cannot bill, or accept credit. Oversize ads are not accepted. Each subscriber is limited to two (2) identical ads in any given issue.
Advertising text and payment must reach us 60 days in advance of publication. For example, advertising copy for the March issue (mailed in February) must be in our hands on January 1. The publisher reserves the right to refuse a questionable or not applicable advertisement. Mail advertisements to: KILOBAUD CLASSIFIED, Kilobaud, Peterborough, NH 03458. Do not include any other material with your ad, as it may be delayed.

Stock Market, Commodity Time Series analysis programs. Exchange with others who have worked in this area on computers. Bob Lemeer, PO Box 29, Brooklyn NY 11230.

PET Owners: Perfect your system before trying Las Vegas. Blackjack, baccarat, roulette, craps. Full LV casino rules & options give maximum reality. Full screen graphics. All on 1 cassette. $25. CMS, 431 Monte Vista, Dallas TX 75223.

Augat w/w boards, 24-16 pin cap. or any comb. of other sizes. 36 dual w/w edge conn. $8, no ICs, $9 w/ICs $10 w/ICs & edge conn. ppd. in U.S. R. Savilonis, 334 Columbia St. Cambridge MA 02141, (617) 492-2095 5-7 PM EST.

HP-97 prog/printing calc. Top-of-line. Orig. $750, asking $475. P. Davis (201) 885-1220 (days).

Sophisticated 2-player football simulation on cassette for Radio Shack TRS-80 Level I BASIC. Uses graphics to draw field, move ball, etc. $12. Dr. S. Harter, Box 17222, Tampa FL 33682.

TRS-80 Sort: in BASIC I, 4K min. Sorts recs in memory on 1 or 2 numeric fields in asc. or desc. seq. Input from tape, k.b. or both. Output to tape or video. Can be subroutine. $10 on tape w/doc. To: Software, Box 6153, Syracuse NY 13217.

PET Software: Game and demo programs written in BASIC for 4K PET. Includes blackjack, children's math, biorhythm and 15 others. On cassette, ready to load and run, plus written descrip. $15.95. R. Schmidt, 14 Tinker Rd., Nashua NH 03060.

PET users need info? Getting started with your PET workbook, $4. Workbook has helpful info, sample programs, exercises and fundamental features of PET BASIC. Send SASE for details. D. Smith, Box 921, Los Alamos NM 87544.

Heathkit H9 video terminal assembled, $400. *Kilobaud* issues 1 to 15, $25. *Byte* issues 3 (Nov 75) to 16 (Dec 76) in binder, $35. *Byte* Jan 77 to date, $15. First money order. T. Halldorson, 44 Ctr. Grove Rd., Apt D-13, Randolph NJ 07801.

Use Kilobaud Classified

Contest!

Another month has gone by, and the votes have been counted. The article winner for the wintry month of February is Dr. Mark Boyd, author of "Interfacing Tips" on page 72.

Choice-of-a-book-from-the-Book Nook winner is Larry Nelson of Marion IN.

To both Mark and Larry, we offer congratulations and best wishes.

And to all of our readers who are responding enthusiastically with their votes, we also offer congratulations, best wishes and good reading.

Keep voting!

NEW PRODUCTS

(from page 15)

passage. Kinged pieces are identified on the display and messages appear at the right of the board relating to each move.

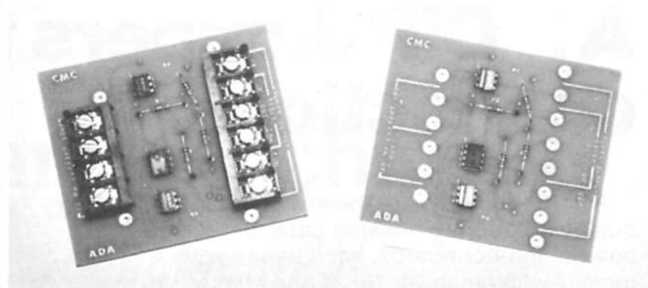
Included with Video Checkers is a 9-page instruction book. The game's author has invited purchasers of the cassette to add enhancements to the program. Therefore, an entire program listing is included.

RS-232 ADApter for KIM

Connecting an RS-232 terminal to KIM is easy with ADA. This ADApter converts KIM's 20-mA current-loop port to an RS-232 port. ADA does not affect the baud rate and uses standard power supplies. The unit comes complete with instructions, is assembled and tested and measures 3 x 3½ x 1 inches. KIM is isolated from the RS-232 device by optoisolators.

ADA sells for $24.50 with drilled, plated-through solder pads for all connections, or for $29.50 with barrier strips and screw terminals.

CONNECTICUT microCOMPUTER, 150 Pocono Road, Brookfield, CT 06804.



CmC's ADA.

This Is the Electric Pencil

The Electric Pencil is a character-oriented word-processing system. This means that text is entered as a continuous string of characters and manipulated as such, allowing the user freedom and ease in the movement and handling of text. Since lines are not delineated, any number of characters, words, lines or paragraphs may be inserted or deleted anywhere in the text. The typing of carriage returns as well as word hyphenation is not required since each line of text is formatted automatically. Whenever text is inserted or deleted, existing text is pushed down or pulled up in a wraparound fashion. Everything appears on the video display screen as it occurs, which eliminates any guesswork.

When text is printed, The Electric Pencil automatically inserts carriage returns where they are needed. Numerous combinations of line length, line spacing and page spacing allow for any form to be handled. Pages may be numbered as well as titled.

Minimum system hardware requirements: (Stock system—other versions available by special order) SOL-20/SOLOS (or equivalent: system w/ VDM-1 and CUTS cassette), 8K of memory starting at location 0000, Printer (TTY, Selectric, etc.), Video Display Monitor, Cassette recorder.

The Electric Pencil is available from stock at: Interactive Computers, 7646 1/2 Dashwood, Houston TX 77036.

Microprocessor Protection

Lightning and heavy-duty electrical equipment often create power-line surges and transients. These can cause extensive damage to valuable microprocessors and peripherals.

Electronic Specialists announces a line-cord transient suppressor that will absorb repeated power surges to protect delicate equipment.

Available in 2-prong plug/socket ($11.50) or 3-prong plug/socket ($14.50), these units are also available with integral power-line hash filtering.

Electronic Specialists, Box 122, Natick MA 01760.

Electronic Specialists' transient suppressor.

KB CALENDAR

Santa Barbara CA

The Southern California Swapfest is being held in Santa Barbara on July 1. John Craig says, "We're going to have an old-fashioned, down-to-earth, good-time swap meet for computer hobbyists, electronics experimenters and hams from all over California."

Atlanta GA

The 16th Annual Convention of the Association for Educational Data Systems will be held in Atlanta GA, May 15-19, 1978. For further information, contact: Dr. James E. Eisele, Office of Computing Activities, University of Georgia, Athens GA 30602.

Washington DC

Amateur Computing 78 microcomputer festival will be held July 22-23 at the Sheraton National Motor Hotel, Arlington VA.

"REMark" is published quarterly

BULK RATE U.S. Postage PAID Heath Company

Two Heathkit User Organizations??

You may have noticed that there are two entirely separate organizations for Heathkit computer users. HUG is the official Heath sponsored organization. The second organization publishes a periodic news letter called BUSS. This second organization is not affiliated with Heath Company in any way. Neither is it approved, sanctioned, or recommended. Heath Company bears no responsibility for the material it publishes or the advice it gives. The official, inside word will always come through HUG. If you want the latest word on new products, software updates, and other juicy news, keep reading REMark.

P.O. Box 4430M Santa Clara, CA 95054
For will call only: (408) 988-1640
2996 Scott Blvd.

Quest ELECTRONICS

Q3

Same day shipment. First line parts only. Factory tested. Guaranteed money back. Quality IC's and other components at factory prices.

INTEGRATED CIRCUITS

7400TTL	
7400N	.17
7402N	.17
7404N	.19
7409N	.23
7410N	.17
7414N	.63
7420N	.17
7422N	1.39
7430N	.20
7442N	.50
7445N	.69
7447N	.60
7448N	.69
7450N	.17
7474N	.29
7475N	.49
7485N	.88
7489N	2.00
7490N	.43
7492N	.43
7493N	.43
7495N	.69
74100N	.90
74107N	.29
74121N	.34
74123N	.59
74125N	.39
74145N	.69
74150N	.95
74151N	.69
74154N	1.00
74157N	.69
74161N	.87
74162N	.87
74163N	.87
74174N	.96
74175N	.90
74190N	1.15
74192N	.87
74193N	.85
74221N	1.55
74298N	1.65
74365N	.66
74366N	.66
74367N	.66
74LS00 TTL	
74LS00N	.25
74LS02N	.25
74LS04N	.25
74LS05N	.25
74LS08N	.25
74LS10N	.25
74LS13N	.40
74LS14N	.90
74LS20N	.25
74LS22N	.25

74LS28N	.41
74LS30N	.25
74LS33N	.39
74LS38N	.30
74LS74N	.35
74LS75N	.47
74LS90N	.51
74LS93N	.51
74LS95N	1.89
74LS107N	.35
74LS112N	.35
74LS113N	.35
74LS132N	.72
74LS136N	.28
74LS151N	.67
74LS155N	.67
74LS157N	.67
74LS162N	.91
74LS163N	.91
74LS174N	.95
74LS190N	1.06
74LS221N	1.95
74LS258N	.67
74LS367N	.89
LINEAR	
CA3045	.90
CA3046	.67
CA3081	1.80
CA3082	1.90
CA3089	2.95
LM301AN/AH	.35
LM305H	.87
LM307N	.35
LM308N	.89
LM309H	1.15
LM309K	.95
LM311H/N	.90
LM317T/K	2.92
LM318	1.35
LM320K-5	1.35
LM323K-5	6.95
LM320K-12	1.35
LM320K-15	1.35
LM320T-5	1.60
LM320T-8	1.60
LM320T-12	1.60
LM320T-15	1.60
LM324N	1.15
LM339N	1.55
LM340K-5	1.10
LM340K-8	1.10
LM340K-12	1.10
LM340K-15	1.10
LM340K-24	1.10
LM340T-5	1.10
LM340T-8	1.10

LM340T-12	1.10
LM340T-15	1.10
LM340T-18	1.10
LM340T-24	1.10
LM343H	4.50
LM370	1.15
LM377	4.50
LM379	5.00
LM380N	1.00
LM381	1.60
LM382	1.60
LM703H	.40
LM709H	.28
LM723H/N	.35
LM733N	.67
LM741CH	.35
LM741N	.35
LM747H/N	.62
LM748N	.35
LM1303N	.82
LM1304	1.10
LM1305	1.27
LM1307	2.00
LM1310	2.75
LM1458	.47
LM1800	1.75
LM1812	7.50
LM1889	3.00
LM2111	1.75
LM2902	1.50
LM3900N	.60
LM3905	1.75
LM3909N	.61
MC1458V	.50
NE540L	2.89
NE550N	.65
NE555V	.43
NE556A	.79
NE565A	1.00
NE566V	1.15
NE567V	1.20
78L05	.60
78L08	.60
79L05	.70
78M05	.85
75108	1.75
75491CN	.50
75492CN	.55
75494CN	.89
A to D CONVERTER	
8038B	4.50
8700CJ	13.95
8701CN	22.00
8750CJ	13.95
LD130	9.95
9400CJV/F	7.40
ICL7103	9.50

CMOS	
CD34001	.50
CD4000	.16
CD4001	.21
CD4002	.21
CD4006	1.10
CD4007	.21
CD4008	.21
CD4009	.39
CD4010	.39
CD4011	.21
CD4012	.21
CD4013	.36
CD4014	.86
CD4015	.86
CD4016	.36
CD4017	.94
CD4018	.94
CD4019	.21
CD4020	1.02
CD4021	1.02
CD4022	.86
CD4023	.21
CD4024	.75
CD4025	.21
CD4026	1.51
CD4027	.36
CD4028	.79
CD4029	1.02
CD4030	.21
CD4035	1.02
CD4040	1.02
CD4042	.71
CD4043	.63
CD4044	.63
CD4046	1.67
CD4049	.36
CD4050	.36
CD4051	1.13
CD4060	1.42
CD4066	.71
CD4068	.40
CD4069	.40
CD4070	.40
CD4071	.21
CD4072	.21
CD4073	.21
CD4075	.21
CD4076	1.75
CD4078	.40
CD4081	.21
CD4082	.21
CD4116	.47
CD4490	5.50
CD4507	1.00
CD4508	4.25
CD4510	1.02

CD4511	.94
CD4515	2.52
CD4516	1.10
CD4518	1.02
CD4520	1.02
CD4527	1.51
CD4528	.79
CD4553	5.75
CD4566	2.25
CD4583	4.50
CD4585	1.10
CD40192	3.00
74C00	.28
74C04	.33
74C10	.28
74C14	2.10
74C20	.28
74C30	.28
74C48	2.95
74C74	.75
74C76	1.40
74C90	1.15
74C93	1.40
74C154	3.00
74C160	1.44
74C175	2.00
74C192	2.40
74C221	2.75
74C905	3.00
74C906	1.50
74C914	1.95
74C922	7.50
74C923	7.50
74C925	10.50
74C926	10.50
74C927	10.50
INTERFACE	
8095	.65
8096	.65
8097	.65
8098	.65
8T09	1.25
8T10	4.50
8T13	3.00
8T20	5.50
8T23	3.10
8T24	3.50
8T25	3.20
8T26	1.69
8T28	2.75
8T97	1.69
8T98	1.69
MOS/MEMORY RAM	
2101-1	4.50
2102-1	1.28
2102AL-4	1.60
21F02	1.85
2104A-4	3.95

2107B	4.00
2111-1	7.00
2112-2	7.90
MK4116	34.95
2513B	8.75
21L02-1	1.49
MM5262	.40
MM5320	5.95
MM5330	5.94
PD411D-3	4.00
PD411D-4	5.00
P5101	13.95
4200A	15.95
82S25	2.90
91L02A	2.50
HD0165-5	6.95
MM57100	9.50
GIAY38500-1	13.95
MCM6571A	15.00
9368	3.00
CLOCKS	
MM5309	3.00
MM5311	3.60
MM5312	4.80
MM5313	3.60
MM5314	3.90
MM5315	4.00
MM5316	5.00
MM5318	8.95
MM5369	2.10
MM5841	14.45
MM5865	7.95
CT7001	5.80
CT7002	8.95
CT7010	8.95
CT7015	7.25
MM5375AA/N	3.90
MM5375AB/N	4.90
7205	16.50
DS0026CN	3.75
DS0056CN	3.75
MM53104	2.50

IC SOCKETS
Solder Tin Low Profile

PIN	1 UP	PIN	1UP
8	.15	24	.36
14	.18	28	.43
16	.20	36	.58
18	.27	40	.61
22	.35		

3 level wire wrap gold 14 pin .35 16 pin .39
2 level 14 pin ww .25

MICROPROCESSOR	
6800	24.50
8080A with data	11.50
Z80	29.95
8212	3.50
8214	8.00
8216	3.85
8224	3.50
8228	6.25
8251	11.50
8255	10.75
CDP1802CD	19.95
CDP1802D	25.00
CDP1861	12.95
6820	12.00
6850	15.00
6502	24.50
UART/FIFO	
AY5-1013	5.50
AY5-1014	7.50
3341	6.95
PROM	
1702A	4.95
N82S23	2.95
N82S123	3.50
N82S126	3.75
N82S129	3.75
N82S131	3.75
N82S136	8.75
N82S137	8.75
2708	16.50
DM8577	2.90
8223	2.90
2716	36.00

IC Test Clips

	1	10
Red	.55	.47
Black	.55	.47

Keyer 8043 14.50
comp. w/spec./sock.

CONNECTORS	
44 pin edge	2.00
100 pin edge	4.50
100 pin edge WW	5.25
TRANSISTORS	
2N2222A	.18
2N3904	.18
2N3906	.18
2N3055	.69

CRYSTALS

1 MHz	4.50		
2 MHz	4.50		
4 MHz	4.25	2.097152 MHz	7.75
5 MHz	4.25	2.4576 MHz	7.50
10 MHz	4.25	3.2768 MHz	7.50
18 MHz	3.90	5.0688 MHz	4.50
20 MHz	3.90	5.185 MHz	4.50
32 MHz	3.90	5.7143 MHz	4.50
32768 Hz	4.00	6.5536 MHz	4.50
1.8432 MHz	4.50	14.31818 MHz	4.25
3.5795 MHz	1.20	18.432 MHz	4.50
2.0100 MHz	1.95	22.1184 MHz	4.50

RESISTORS ¼ watt 5%
10 per type .03 1000 per type .012
25 per type .025 350 piece pack
100 per type .015 5 per type 6.75

KEYBOARDS
Hex keyboard $10.95. Fully encoded w/ PC board, parts and instructs. $29.95
53 key ASCII keyboard kit 55.00
Fully assembled 65.00 Enclosure 14.95

LEDS	
Red T018	.15
Green, Orange, Yellow T018	.20
Jumbo Red	.20
Green, Orange, Yellow Jumbo	.25
Cliplite LED Mounting Clips	8/$1.25

(specify red, amber, green, yellow, clear)

CONTINENTAL SPECIALTIES in stock
Complete line of breadboard test equip.
MAX-100 8 digit Freq. Ctr. $128.95

OK WIRE WRAP TOOLS in stock
Portable Multimeter $18.00

DIGITAL THERMOMETER $46.50
Batt. oper. General purpose or medical 32°–230°F. Disposable probe cover ± .2° accuracy. Comp. assy. in compact case.

COMPUTER BOARD KITS	
8K RAM Board Kit	$134.95
4K EPROM Kit	114.95
I/O Board Kit	44.50
Extender Board w/connector	12.50
Video Interface board kit	125.00
16K EPROM board kit w/o PROMS	74.50
16K Static RAM board kit	395.00
North Star Floppy Disk Kit	$665.00
Additional Drive Kit	415.00

SPECIAL PRODUCTS	
MM5865 Stopwatch Timer	9.00
PC board	7.50
Switches Mom. Pushbutton	.27
3 pos. slide	.25
Encoder HD0165-5	6.95

3 Digit Universal Counter Board Kit
Operates 5–18 Volt DC to 5 MHz typ. .125" LED display **10.50**
Voice actuated switch .50

Paratronics 100A Logic Analyzer Kit **$199.00**
Model 10 Trigger Expander Kit $299.00
Model 150 Bus Grabber Kit $369.00

TRANSFORMERS	
12 Volt 300 ma transformer	1.25
12.6V CT 600 ma	3.75
12V 250 ma wall plug	2.95
12V CT 250 ma wall plug	3.50
24V CT 400 ma	3.95
10V 1.2 amp wall plug	4.85

DISPLAY LEDS			
MAN1	CA	.270	2.90
MAN3	CC	.125	.39
MAN72/74	CA/CA	.300	1.00
DL704	CC	.300	1.25
DL707/DL707R	CA	.300	1.00
DL727/728	CA/CC	.500	1.90
DL747/750	CA/CC	.600	1.95
DL750	CC	.600	1.95
FND359	CC	.357	.70
FND500/507	CC/CA	.500	1.35
FND503/510	CC/CA	.500	.90
FND800/807	CC/CA	.800	2.20
3 digit Bubble			.60
4 digit bubble			.80
DG8 Fluorescent			1.75
DG10 Fluorescent			1.75
5 digit 14 pin display			1.00
NSN33M 3 digit 8 pin			.39
NSN69 9 digit display			.60
7520 Clairex photocells			.39
TIL311 Hex			9.50

5 function digital DC Calendar clock $54.95
4" green LED display, auto. dim. Pedestal mount 2.7" by 2.7". Beautiful case.

Sinclair 3½ Digit Multimeter
Batt. oper. 1mV and .1NA resolution. Resistance to 20 meg. 1% accuracy. Small, portable, completely assem. in case. 1 yr. guarantee. Best value ever! **$59.95**

Not a Cheap Clock Kit $14.95
Includes everything except case. 2-PC boards. 6-.50" LED Displays. 5314 clock chip, transformer, all components and full instructions. **Same clock kit with .80" displays. $21.95**

Digital Temperature Meter Kit
Indoor and outdoor. Automatically switches back and forth. Beautiful. .50" LED readouts. Nothing like it available. Needs no additional parts for complete, full operation. Will measure −100° to +200°F, air or liquid. Very accurate. Complete instructions. **$39.95**

NiCad Batt. Fixer/Charger Kit
Opens shorted cells that won't hold a charge and then charges them up, all in one kit w/full parts & instruc. **$7.25**

RCA Cosmac VIP Kit 275.00
Video computer with games and graphics.

'78 IC Update Master Manual
1978 IC Update Master Manual **$30.00** Complete IC data selector 2175 pg. Master reference guide. Over 42,000 cross references. Free update service through 1978. Domestic postage $3.50. Foreign $6.00. Final 1977 Master closeout **$15.00**

New Cosmac Super "ELF"
RCA CMOS expandable microcomputer w/HEX keypad input and video output for graphics. Just turn on and start loading your program using the resident monitor on ROM. Pushbutton selection of all four CPU modes. LED indicators of current CPU mode and four CPU states. Single step op. for program debug. Built in pwr. supply. 256 Bytes of RAM, audio amp. & spkr. Detailed assy. man. w/PC board & all parts fully socketed. Comp. Kit **$106.95** High address display option **8.95**; Low address display option **9.95**; Custom hardwood cab.; drilled front panel **19.75** Nicad Battery Backup Kit w/all parts **4.95** Fully wired and tested in cabinet **151.70** 1802 software xchng. club; write for info.

4K Elf Expansion Board Kit with Cassette I/F $79.95
Available on board options: 1K super ROM monitor **$19.95** Parallel I/O port **$7.95** RS232 I/F **$3.50** TTY 20 ma I/F **$1.95** S-100 Memory I/F **$5.30**

Tiny Basic for ANY 1802 System
Kansas City Standard Cassette $10.00
On ROM Monitor $38.00
Super Elf owners take 30%

Original Cosmac "ELF" kit
All parts and instructs. **$89.50**
Board only **14.95**

Video Modulator Kit $9.95
Convert your TV set into a high quality monitor without affecting normal usage. Complete kit with full instructions.

60 Hz Crystal Time Base Kit $4.75
Converts digital clocks from AC line frequency to crystal time base. Outstanding accuracy. Kit includes: PC board, MM5369, crystal, resistors, capacitors and trimmer.

Clock Calendar Kit $23.95
CT7015 direct drive chip displays date and time on .6" LEDS with AM-PM indicator. Alarm/doze feature includes buzzer. Complete with all parts, power supply and instructions, less case.

2.5 MHz Frequency Counter Kit
Complete kit less case **$37.50**

30 MHz Frequency Counter Kit
Complete kit less case **$47.75**
Prescaler Kit to 350 MHz **$19.95**

Stopwatch Kit $26.95
Full six digit battery operated. 2–5 volts. 3.2768 MHz crystal accuracy. Times to 59 min., 59 sec., 99 1/100 sec. Times std., split and Taylor. 7205 chip, all components minus case. Full instruc.

Auto Clock Kit $15.95
DC clock with 4-.50" displays. Uses National MA-1012 module with alarm option. Includes light dimmer, crystal timebase PC boards. Fully regulated, comp. instructs. Add $3.95 for beautiful dark gray case. Best value anywhere.

TERMS: $5.00 min. order U.S. Funds. Calif residents add 6% tax. BankAmericard and Master Charge accepted. Shipping charges will be added on charge cards.

FREE: Send for your copy of our NEW 1978 QUEST CATALOG. Include 24¢ stamp.

SOLID CAREER OPENING WITH HEATH

You know us for our quality Heathkits – and for our reputation for "doing things right." Our progress story goes far back, and prospects are very favorable for the years ahead.

SOFTWARE DOCUMENTATION WRITER

An effective, proven technical writer needed to write Heathkit computer systems operations manuals and applications procedures. A B. Sc. CS or equivalent is required. Candidates must be familiar with LSI – 11&PDP–11, software and applications programming, higher level languages, and operating systems, principles and techniques.

Salaries are competitive, with excellent benefits additional. Our ideal location is a pleasant, small community on Lake Michigan, 90 minutes from Chicago.

Please send resume in confidence, or phone collect to Ken Smith, (616) 982-3673.

HEATH
Schlumberger

HEATH COMPANY H5
49022
Benton Harbor, Michigan
An Equal Opportunity Employer M/F

COMPUTER CLUBS!

Do you need to start a club library?

Start with the BEST!

Have the secretary of your club send us your current club roster, complete with names, addresses and zip codes and we'll send you one of your choice of the following books, . . . FREE!

Hobby Computers Are Here
The New Hobby Computers
IC Test Equipment, the new Vol. IV of the 73 Test Equipment Library

(limit one book per club)

Kilobaud Clubs
Peterborough, NH 03458

MICROCOMPUTERS, INC.

SMALL BUSINESS SYSTEMS
DEDICATED MICROPROCESSOR
APPLICATIONS

539 Amherst St.
Nashua, N.H. 03060 Tel. (603) 889-1646

M16

The All New!

Personal & Small Business Computer Expo "South"

Plan To Exhibit

Plan To Attend

May 19–21, 1978, Exposition Park
Orlando, Florida

For Details Call Or Write:
Felsburg Associates, Inc. (301) 262-0305
P.O. Box 735, Bowie, Md., 20715

THE COMPUTER CORNER

White Plains Mall, Upper Level
200 Hamilton Ave.
White Plains NY 10601
Phone: (914) WH9-DATA

Near Bronx River Parkway & Cross Westchester Expressway. Plenty of parking.

"The S100 Bus stops at White Plains" with one of the largest collections of boards compatible with the Altair Bus (also IMSAI) in the greater NY area.

You've read about the Sol-20, now come up and see it. We carry Processor Tech, Polymorphic, IMSAI, North Star, TDL, Blast Master and Pickles and Trout.

GOOD PRICE AND SERVICE
10-6 Mon.-Sat.
Thurs. till 9

C-28

THE COMPUTER CORNER

SURPLUS ELECTRONICS

IBM SELECTRIC® BASED I/O TERMINAL (USED) $695.00

- Tape Drives • Cable
- Cassette Drives • Wire
- Power Supplies 12V15A, 12V25A, 5V35A Others, • Displays
- Cabinets • XFMRS • Heat Sinks • Printers • Components

Many other items
Write for free catalog
WORLDWIDE ELECT. INC.
10 FLAGSTONE DRIVE
HUDSON, N.H. 03051
Phone orders accepted using VISA or MC. Toll Free 1-800-258-1036
In N.H. 603-885-3705

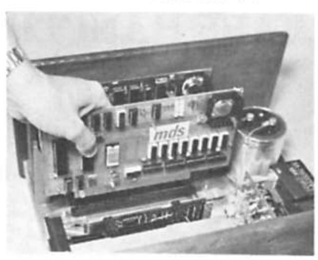
W16

LOOKING FOR THE BEST COMPUTER ON THE MARKET?
WHETHER YOU ARE JUST STARTING OR WANT TO UPGRADE YOUR PRESENT SYSTEM, WHY NOT HAVE —

THE BEST OF BOTH WORLDS

6800 COMPUTING POWER (With the new 6802!) — AND — S-100 COMPATIBILITY

YOU CAN HAVE *BOTH* WITH THE MD690 CENTRAL PROCESSOR FROM MDS



AVAILABLE AS SINGLE BOARD OR COMPLETE SYSTEM:

MD690 CENTRAL PROCESSOR BOARD

Single board includes 6802 processor, 1K monitor PROM (Standard 6800 compatible), 1152 bytes of RAM, on-card 2400 baud cassette interface, parallel / serial I/O port — Complete instructions — Ready to operate — Just plug into any S-100 bus — Interfaces easily with most keyboards and memory-mapped video cards. KIT $198 ASSEMBLED $269

MDS-2 SYSTEM

Complete system includes case, MD690 processor board, power supply, video / graphics board, motherboard with two edge connectors (room for 6), and keyboard with case. KIT $579 ASSEMBLED $798

UNBEATABLE PRICESUNEXCELLED PERFORMANCE***

Either board or complete system is supported by extensive MDS documentation and software — every effort has been made to make this micro easy to understand, easy to use, easy to buy.

NEW!

ALL MDS SOFTWARE NOW AVAILABLE BY TELEPHONE DATA LINK!

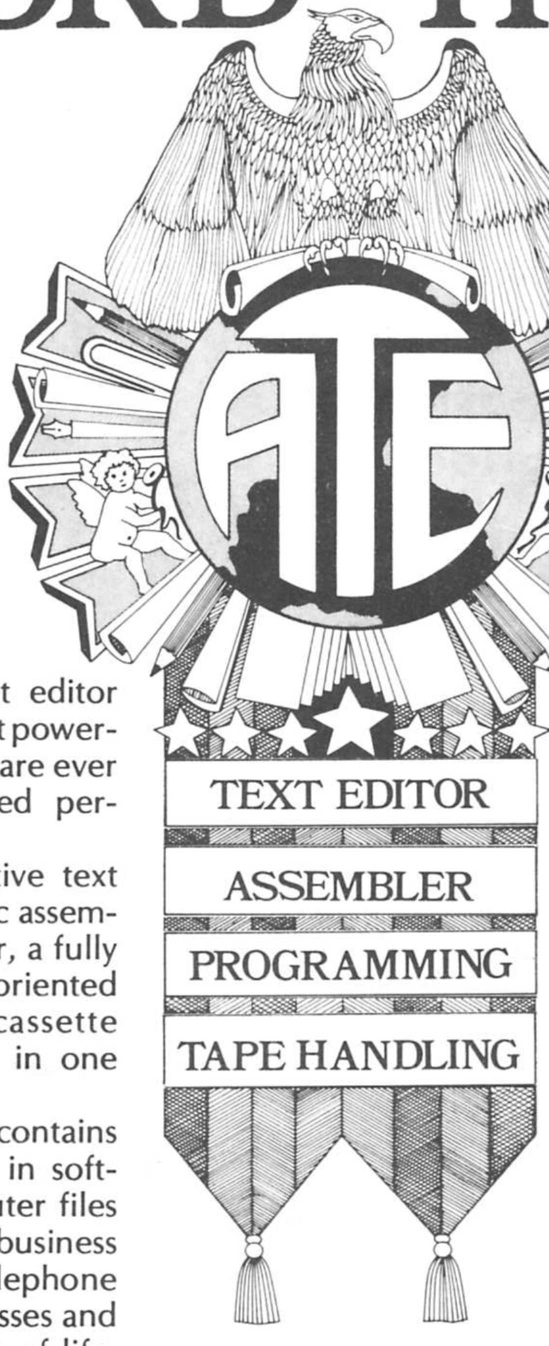
TO ORDER — SEND CHECK OR MONEY ORDER TO: **LAKE ELECTRONICS** L14
Dept. C, Box 45058
Los Angeles, CA 90045
Cal.Res. Add 6% Sales Tax

The Computer Mart
Toronto, Canada

YOUR NUMBER ONE SOURCE FOR MICROS AND PERIPHERALS.
PROFESSIONAL SERVICE AND GUIDANCE.
CATALOG $2.00

1543 Bayview Avenue,
Toronto, Ontario M4G3B5
(416) 484-9708

WORLD'S RECORD HOLDER



ATE* assembler/text editor by Soft Corp is the most powerful text-oriented software ever created for 8080-based personal computers.

ATE* is an interactive text editor, a fully symbolic assembler, a system monitor, a fully programmable text-oriented language, and a cassette operating system—all in one 4K package.

In short, ATE* contains everything you need in software to create computer files for your personal or business records of names, telephone numbers, dates, addresses and the other textual facts of life.

Then, using ATE's* powerful programming language, you can build your own programs to put your files to work in exciting, highly personalized applications.

ATE* contains so many features to make text-oriented operations simple, logical and efficient that you'll have to see it to believe it. Editing by character, string or line. Interactive and programmable editing. Printing, editing and assembling object code in your choice of octal, hex or decimal. And so much more we can't even begin to list ATE's* full array of features.

Ask to see ATE* demonstrated at your nearest computer store. Or ask your local computer store to order it for you.

ATE* on Kansas City Standard 300 baud cassette with Users' Manual, $25.†

If unavailable locally, order direct from Thinker Toys™, 1201 10th St., Berkeley, CA 94710. Call (415) 527-7548 (10- 4 PST). Mail and phone orders add $3; Cal. res. add tax.

A product of Soft Corp for

Thinker Toys™

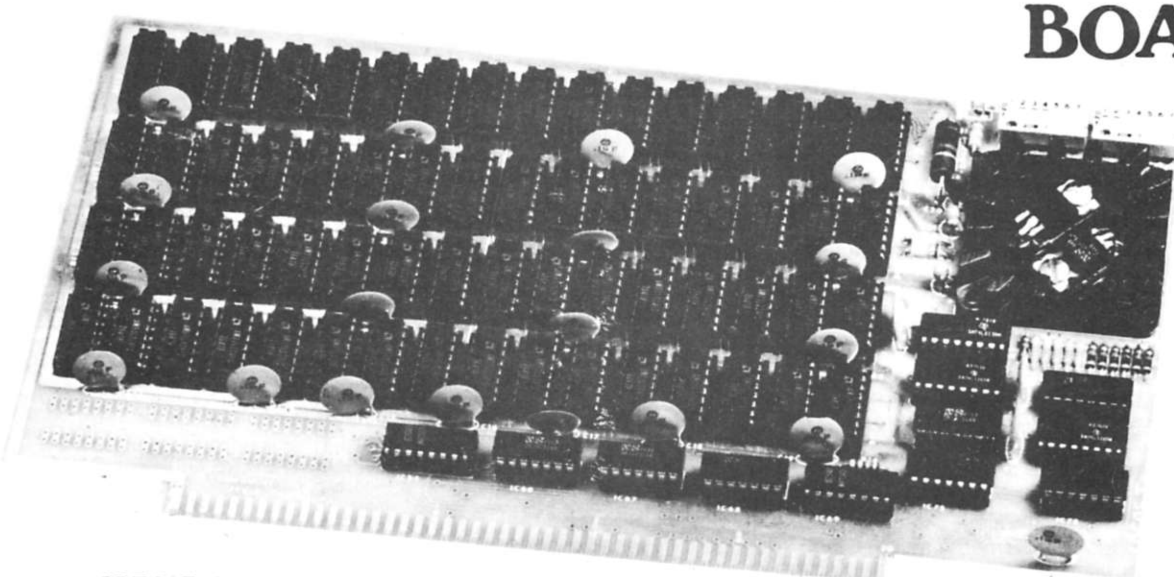
1201 10th Street Berkeley, CA 94710

T2

*ATE is a tm of Soft Corp. †Deutsche übersetzung von computershop Gmbh, D-7800, Freiburg im Breisgau, Adelhauserstrasse 29, W. German

ADVANCED COMPUTER PRODUCTS

LOGOS I
8K STATIC MEMORY BOARD



SPECIFICATIONS:

System Compatibility:	S-100 buss compatible. Altair/Imsai compatible.
Memory Protect:	Hardware memory protect circuit features protection of board as one 8K byte block, two 4K blocks, four 2K blocks, eight 1K blocks sixteen 512 byte blocks, or thirty-two 256 byte blocks, or disable entirely all options are dipswitch selectable.
Addressing:	Addressing on any 1K boundary.
Wait States:	No Wait States
Buffering:	Truly buffered. Address lines and Data-in and Data-out fully Buffered. Key feature as most 8K memory boards are not fully buffered.
Speed:	Logos I — 450 ns access. Allows board to run at full speed with no wait states. Logos IZ — 250 ns access. Allows board to run at full 4MHz Z-80 speed.
Support Chips:	Low power Schottky (where applicable)
PC Board	Hi-grade gloss epoxy with plated thru holes, gold-plated edge connector contacts, solder-mask, with silk screen.
Power:	+8v, 1.5 Amps typical
Battery Back-up:	Circuitry on board

ORDER INFORMATION

Logos I 8K Kit $125.95 Buy (4) Units reduce to $117.00
Logos I 8K Assembled & Tested $179.95
Logos I 8K-Z (250ns.) Kit $149.95
Logos I 8K-Z Assembled & Tested $199.95

Imsai Compatible Wirewrap Connector $4.75
Imsai Compatible Solder-tail Connector $4.75
Altair Compatible Wirewrap Connectors $5.95
Altair Compatible Solder-tail Connectors $5.95

ADVANCED COMPUTER PRODUCTS INC.
Irvine, CA 92713 • P. O. Box 17329 • (714) 558-8813 • TELEX/TWX: 910-595-1565

S.D. COMPUTER PRODUCTS

AN EMPIRE IND. CO.

S2

P.O. BOX 28810K DALLAS, TEXAS 75228

EXPANDABLE EPROM BOARD

16K OR 32K EPROM $49.95 W/OUT EPROM

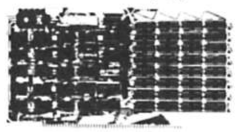
Allows You to Use Either 2708's For 16K of Eprom or 2716's For 32K of Eprom.

KIT FEATURES:

1. All Address Lines & Data Buffered
2. Quality Plated through P.C. Board Including Solder Mask and Silk Screen
3. Selectable Unit States
4. On Board Regulation Provided
5. All Sockets Provided W/Board

WE CAN SUPPLY 450As 2708's AT $11.95 WHEN PURCHASED WITH BOARD.

EXPANDORAM
THE ULTIMATE RAM BOARD
32K FOR $475.00



32K MODEL	
8K	$151.00
16K	259.00
24K	367.00
32K	475.00

64K MODEL	
16K	$281.00
32K	519.00
48K	757.00
64K	995.00

THE 32K VERSION USES THE MOSTEK MK4115 RAM AND HAS 8K BOUNDARIES AND PROTECTION & UTILIZES DIP SWITCHES. P. C. BOARD COMES WITH SOCKETS FOR 32K OPERATION

BUY A S100 COMPATIBLE RAM BOARD AND UPGRADE THE SAME BOARD TO A MAXIMUM OF 65K MEMORY IN STEPS OF EITHER 8K OR 16K AT YOUR OPTION BY MERELY PURCHASING MORE RAM CHIPS FROM SD COMPUTER PRODUCTS.

THE 64K VERSION USES THE MOSTEK MK4116 RAM AND HAS 16K BOUNDARIES AND PROTECTION & UTILIZES DIP SWITCHES. P. C. BOARD COMES WITH SOCKETS FOR 64K OPERATION

LOOK AT THE FEATURES WE HAVE BUILT INTO THE EXPANDORAM!

- MEMORY ACCESS TIME IS 375 ns
- MEMORY CYCLE TIME IS 500 ns
- POWER REQUIREMENTS ARE:
  8 VDC 400 MA DC
  18 VDC 400 MA DC
  —18 VDC 30 MA DC
- ON BOARD INVISIBLE REFRESH
- NO WAIT STATES REQUIRED
- NO CYCLE STEALING NEEDED
- ON BOARD REGULATION
- CONTROL, DATA & ADDRESS INPUTS UTILIZE LOW POWER SCHOTTKY DEVICES
- DESIGNED TO WORK WITH Z-80, 8080, 8085, CPU's

ADD $50.00 TO ABOVE PRICES FOR FULLY ASSEMBLED AND TESTED BOARDS

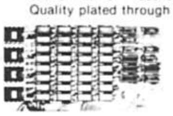
Low Cost Cassette Interface Kit

$19.95

Features: Play and record K.C. Standard 2400/1200 Hz tapes, 300 Baud, TTL I/O Compatible, Phase Lock Loop, Both 22 Pin Connector and 8 Pin Molex Connector. Comes partially assembled. Oscillator and phase lock loop pre tuned to K.C. Standard. Selector switch sends cassette data or auxiliary input data to microprocessor. LED indicates logic 1 level.

4K LOW POWER RAM KIT
The Whole Works - $79.95

Full Buffered - on board regulated - reduced power consumption utilizing low power 21L02-1 500ns RAMS - Sockets provided for all IC's. Quality plated through PC board.



*Add $10.00 for 250ns RAM operation.

8K LOW POWER RAM
$159.95

FULLY ASSEMBLED AND TESTED. NOT A KIT.

Imsai — Altair — S-100 Buss compatible, uses low power static 21L02-500ns fully buffered on board regulated, quality plated through PC board, including solder mask. 8 pos. dip switches for address select.

*Add $30.00 for 250ns RAM operation

Z-80 CPU BOARD KIT
Complete Kit $139.

CHECK THE ADVANCED FEATURES OF OUR Z-80 CPU BOARD: Expanded set of 158 instructions, 8080A software capability, operation from a single 5VDC power supply; always stops on an M1 state, true sync generated on card (a real plus feature!), dynamic refresh and NMI available, either 2MHZ or 4MHZ operation, quality double sided plated through PC board; parts plus sockets provided for all IC's. *Add $10. extra for Z-80A chip which allows 4MHZ operation.

NEW FROM S.D.
"VERSAFLOPPY"™ KIT
THE VERSATILE FLOPPY DISK CONTROLLER
ONLY $149.00

FEATURES: IBM 3740 Soft Sectored Compatible. S-100 BUS Compatible for Z-80 or 8080. Controls up to 4 Drives (single or double sided). Directly controls the following drives:

1. Shugart SA400/450 Mini Floppy
2. Shugart SA800/850 Standard Floppy.
3. PERSCI 70 and 277.
4. MFE 700/750.
5. CDC 9404/9406.

34 Pin Connector for Mini Floppy. 50 Pin Connector for Standard Floppy. Operates with modified CP/M operating system and C-Basic Comptler. The new "Versafloppy" from S.D. Computer Products provides complete control for many of the available Floppy Disk Drives, Both Mini and Full Size. FD1771B-1 Single Density Controller Chip. Listings for Control Software are included in price.

FD 1771B-1 CHIP ALONE $39.95

S. D. SALES NEW
EXPANDABLE EPROM BOARD

16K OR 32K EPROM $49.95 W/OUT EPROM

Allows You to Use Either 2708's For 16K of Eprom or 2716's For 32K of Eprom.

KIT FEATURES:

1. All Address Lines & Data Buffered
2. Quality Plated through P.C. Board Including Solder Mask and Silk Screen
3. Selectable Unit States
4. On Board Regulation Provided
5. All Sockets Provided W/Board

WE CAN SUPPLY 450As 2708's AT $11.95 WHEN PURCHASED WITH BOARD.

INTRODUCING THE SBC-100
(The Z-80 Based, S-100 Single Board Computer)

$349.00

FEATURES:

- No Front Panel Needed
- Z-80 CPU (2 or 4 MHZ)
- 1K RAM
- 4 ROM/PROM Sockets for 4K/8K of Memory
- SYNCHRONOUS/ASYNCHRONOUS Serial I/O with RS-232 and Current Loop Interface and Software
- Programmable Baud Rate
- Parallel Input Port
- Parallel Output Port
- 4 Channel Timer/Counter
- 4 Vectored Interrupts

O. E. M. SPECIAL

ASK ABOUT SPECIAL O.E.M. DISCOUNTS ON THE S.D. "COMPATIBLE SET."

SBC-100 — SINGLE BOARD COMPUTER	$349.00
VERSAFLOPPY™ — FLOPPY DISK CONTROLLER	$149.00
EXPANDORAM — 32K RAM	$475.00

EACH KIT IF PURCHASED SEPARATELY TOTAL $973.00

ORDER ALL 3 KITS TOGETHER FOR

$899.00

This Powerful Threesome Operates Together to Form A Complete Computer for Your System.

Z-80 Programming Manual — IN DEPTH DETAIL OF THE Z-80 CPU MICRO-COMPUTER — **S. D. SALES SPECIAL $9.95**

RAMS

21L02 - 500NS	8/11.50
21L02 - 250NS	8/15.95
2114 — 4K	14.95
1101A — 256	8/$4.00
1103 — 1K	35
MK 4115 - 8K	15.45
74S 200 - 256	3.95

CPU's

Z—80 includes manual	29.95
Z—80A includes manual	34.95
8080A CPU 8 BIT	11.95
8008 CPU 8 BIT	6.95

PROMS

1702A - 1K - 1.5us	3.95 or 10/35.
2708 - 8K - 450ns	14.95
5204 - 4K	7.95
82S129 — 1K	2.50
2708U 8K signetics 650ns	9.95

COUNTER CHIPS

MK50397 6 Digit elapsed timer	8.95
MK50250 Alarm clock	4.99
MK50380 Alarm chip	2.95
MK50395 6 digit up/dn. count.	12.95
MK5002 4 digit counter	8.95
MK5021-Cal. chip sq. root	2.50

S.D. NOW HAS SOFTWARE FOR IT'S CUSTOMERS

CP/M® DISK OPERATING SYSTEM . . . $99.95

CP/M is a powerful disk operating system which has become an industry standard. It is compatible with several disk based FORTRAN and BASICS. This package includes a CP/M diskette (mini or full size) adapted for S.D.'s SBC-100/VERSAFLOPPY EXPANDORAM board set. Complete documentation is included. (® CP/M is a registered trademark of Digital Research Corp., Pacific Grove, CA.

Z-80® DISK BASED ASSEMBLER $69.95

Runs on ANY CP/M based disk system. Assembles the official Zilog-Mostek Mnemonics. Contains extensive set of pseudo-ops. Available on mini or full size diskette.

VERSAFLOPPY™ CONTROL FIRMWARE . . . $24.95

Provides control for VERSAFLOPPY and boots up CP/M. This runs on Z-80, 8080 or 8085 based computers. Available in 2708 or 2758 prom.

SD MONITOR $49.95

Powerful monitor for SBC-100 single board computers. Includes all VERSAFLOPPY control firmware. Comes in 2716 prom. Available in 4-6 weeks.

VERSAFLOPPY DIAGNOSTIC PROGRAM . . . $24.95

Provides routines which are helpful in checking out a disk based system. Available in 2708 or 2758 prom.

SUPER FLOPPY SPECIAL

S. D. SALES' VERSAFLOPPY S-100 CONTROLLER BOARD PLUS SHUGART SA 400 FLOPPY DISK DRIVE INCLUDING CABLE FOR ONLY

$449.00

MICROPROCESSOR CHIPS

8212 - I/O port	3.50
8214 — P.I.C.	12.95
8216 — Non Invert Bus	4.95
8224 — Clock Gen.	4.95
8226 — Invert Bus	3.95
PIO for Z—80	14.95
CTC for Z—80	14.95
8228 Sys. Controller	8.20
8251 Prog. comm. interfact.	10.95
8255 prog. prep. interface	13.50
8820 Dual Line Recr	1.75
8830 Dual Line Dr.	1.75
2513 Char. Gen.	7.50
8838 Quad Bus. Recvr.	2.00
74LS138N — 1/8 decoder	.99
8T97-Hex Tri-State Buffer	1.25
1488/1489 RS232	1.50
TR 1602B Uart	3.95
TR 1853 Uart	8.50
FD 1771B-1	39.95

CMOS

4001	19	4029	99
4002	19	4042	69
4011	19	4047	1.50
4013	32	4049	35
4016	32	4069	23
4017	95	4071	19
4020	97	4076	97
4022	97	14518	1.10
4024	75	14528	85
4027	39	14529	85

CALL IN YOUR BANKAMERICARD (VISA) OR MASTER CHARGE ORDER IN ON OUR CONTINENTAL TOLL FREE WATTS LINE:

1—800—527—3460

Texas Residents Call Collect:

214/271—0022

DEALER INQUIRIES INVITED!

VISA — master charge

(All prices subject to change without prior notice.)

NO COD's. TEXAS RESIDENTS ADD 5% SALES TAX. ADD 5% OF ORDER FOR POSTAGE & HANDLING. ORDERS UNDER $10. ADD 75c HANDLING. FOREIGN ORDERS — U. S. FUNDS ONLY!

California Industrial

Post Office Box 3097 K • Torrance, California 90503

Diablo terminal

In February, when this advertisement was submitted to the publisher, we were negotiating for the purchase of several hundred used Diablo Terminals.

If we are successful in acquiring these units, they will be available in late April for only $850.00. FOB Los Angeles.

All terminals were removed from service in operating condition.

Electronic Entertainment Center

Tennis-Handball Hockey-Smash

Action-packed color entertainment for the whole family. Adjustable skill level controls allow players of all ages to compete in tennis, hockey and handball. This four game entertainment center turns your television into a video playground.

On screen scoring, live action sound and true component color makes this video center an excellent buy at only $24.88. Complete with antenna box and AC adapter.

Color $24.88

TELETYPE® MODEL 43

New from Teletype, the Model 43 is capable of printing 132 ASCII characters per line. Send and receive data at 10 or 30 char. per second. Keyboard generates all 128 ASCII code combinations. RS-232 interface, same as the popular Model 33. Data sheet sent upon request. Manufacturer suggested price $1377.00.

IMMEDIATE DELIVERY $1219

TTL model with NOVATION brand Acoustic Modem. $1419

HEXADECIMAL KEYBOARD

$34.95

Maxi-Switch hexadecimal keyboards are designed for microcomputer systems that require 4-bit output in standard hex code.

Each assembly consists of 16 hermetically sealed reed switches and TTL "one shot" debounce circuitry.

Reliable low friction acetal resin plungers are credited for the smooth operation and long life of this premium keyboard.

Requires single +5 volt supply.

REGULATED POWER SUPPLY

Delivers 5 volts at 8 Amperes along with three other regulated outputs.

This used supply is sold "as is," but we still feel that this is the best buy you'll ever see in a regulated power supply. 75 lbs., Schematics included.

Used $49.50

Shipped Freight Collect

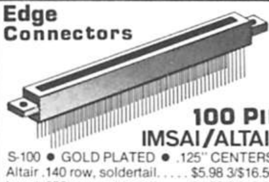
CONNECTORS

RS-232

DB25P male plug & hood $3.95

DB25S female $3.95

Edge Connectors



100 PIN IMSAI/ALTAIR

S-100 • GOLD PLATED • .125" CENTERS

Altair .140 row, soldertail. . . . . $5.98 3/$16.50

Imsai .250 row, soldertail. . . . . $4.98 3/$13.00

3 Level Wire Wrap .250 row. . . $4.98 3/$13.00

SPECIALS

W/W same as above without ears $3.50 3/$10

72 (dual 36) W/W .156" centers. . . $2.50 3/$6

$24.88

SPERRY UNIVAC KEYBOARD

The famous Sperry Univac 1710 Hollerith keyboard assembly is now available from California Industrial for only $24.88. The ideal computer input device for accountants and mathematicians. The numeric keys are placed on the lower three rows to resemble a ten key adding machine. This format allows one handed numeric data entry.

Original cost was $385. Used but guaranteed in excellent condition. Complete with documentation.

Quiet Buss™ S-100 MOTHER

The Quiet Buss from California Industrial is quality engineered. No short cuts have been taken to produce this mother board. Active termination circuitry prevents noise and crosstalk. Manufactured from extra heavy FR-4 epoxy glass. Features 2 ounce double thickness copper traces.

18 SLOT 29.95

FREE

MANUAL GRAPHITE DISPLAY GENERATOR

Modern technology has pioneered the development of this unique character printer. Our Manual Graphite Display Generator has the capability of producing the full upper and lower case ASCII set. Self-contained cursor assembly allows the operator to eliminate erroneously entered information.

Each unit is manufactured to strict tolerances as prescribed by standards set forth by California Industrial. One free with every order.

Scotch® BRAND

$4.98 10 for $45.

DISKETTES

Please specify IBM 3740 series or 32 sector.

also MINIDISKS

Certified Digital CASSETTES

Won't drop a BIT!

$5.50

CALIFORNIA INDUSTRIAL is an Authorized Dealer of Scotch Brand Data Products

JOYSTICK $4.50

3 for $10.00

This joystick feature four 100K potentiometers, that vary resistance proportional to the angle of the stick. Perfect for television games, quad stereo and radio controlled aircraft.

450ns 2708 $9.95

8K UV Erasable MEMORY

5&12 VoLt REGULATORS

$1.19 ea. 10 .97 100 .88

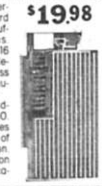
	Negative	Positive
TO-3	LM320K( )	LM340K( )
TO-220	LM320T( )	LM340T( )

S-100 PROTOTYPE BOARD

$19.98

GP100-Maximum design versatility along with standard address decoding and buffering for S100 systems. Room for 32 uncommitted 16 pin IC's, 5 bus buffer & decoding chips, 1 DIP address select switch, a 5 volt regulator and more.

WW100-Wire wrap breadboard, similar to the GP100. Allows wire wrap of all sizes of sockets in any sizes of sockets in any combination. An extra regulator position for multiple voltage applications.



21L02

LOW POWER 450 NS

$1.19

SUPER BUY!

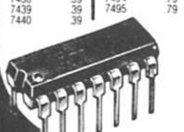
Lowest Price Anywhere

Our low power static RAMs are factory prime. Purchased, on contract, directly from one of California's leading semiconductor manufacturers.

Access time guaranteed faster than 450nS. **Minimum purchase 32 pieces.**

Sorry, credit cards can not be accepted on 21L02's.

7400	.13	7442	.59
7401	.19	7443	.79
7402	.19	7445	.89
7403	.19	7446	.99
7403	.19	7447	.99
7404	.19	7448	.99
7405	.19	7450	.25
7406	.19	7451	.25
7407	.25	7453	.25
7408	.25	7454	.25
7409	.25	7460	.25
7410	.19	7470	.25
7411	.25	7472	.39
7412	.35	7473	.39
7413	.49	7474	.35
7414	.79	7475	.49
7416	.39	7476	.39
7417	.39	7479	3.99
7420	.19	7480	.79
7422	.49	7482	.99
7423	.39	7483	.99
7425	.39	7485	.99
7426	.39	7486	.49
7427	.39	7488	3.40
7428	.49	7489	2.79
7429	.39	7490	.49
7430	.25	7491	.99
7433	.39	7492	.49
7437	.39	7493	.49
7438	.39	7494	.79
7439	.39	7495	.79
7440	.39		



7496	.79
7497	3.99
74100	1.19
74107	.39
74109	.49
74110	1.79
74116	1.99
74120	1.79
74121	.39
74122	.39
74123	.69
74125	.59
74126	.59
74128	.49
74132	.99
74136	.89
74141	.99
74145	.99
74147	2.49
74148	1.99
74150	1.19
74151	.99
74153	.89
74154	.99
74155	.99
74156	1.29
74157	.99
74159	2.99
74160	1.19
74161	.99
74162	1.49
74163	.99
74166	1.19
74167	4.99
74170	2.49
74173	1.49
74174	1.19
74175	.99
74176	.99
74177	.99
74192	.99
74193	.99

CMOS

4001	.25
4002	.25
4006	1.99
4007	.25
4008	1.49
4009	.69
4010	.69
4011	.25
4012	.25
4013	.49
4014	1.49
4015	1.39
4016	.69
4017	1.29
4018	1.69
4019	1.79
4020	1.39
4021	1.49
4022	1.25
4023	.25
4024	1.19
4025	.25
4027	.69
4028	1.25
4029	1.99
4030	.69
4032	.49
4033	1.99
4035	1.69
4040	1.99
4041	1.49
4042	1.49
4043	1.49
4044	1.49
4046	2.49
4047	2.49
4049	.79
4050	.79
4051	1.99

CPU's

8080A	14.95
6800	24.95
Z80	39.95

MEMORY

1702A	4.95
82s23	2.95
82s123	2.95
2102	1.79
2102-1	1.89
21L02	1.19
250nS	1.49

CLOCK's

5314	2.95
5316	4.95
5375	3.95

LM741 $.09

301H	.39	350N	.99
301CN	.39	351CN	.65
302H	1.29	370H	1.29
304H	1.29	370N	1.29
305H	.99	373N	3.19
307H	.49	377N	3.99
307CN	.39	380N	1.39
308H	.99	381N	1.79
308CN	.99	382N	1.79
309H	1.09	NE555v	.49
309K	.99	NE556	1.29
310H	1.19	NE565H	1.49
310CN	1.19	NE565N	1.79
311H	.99	NE566N	1.25
311N	.99	703CN	.45
312H	1.99	709H	.39
318H	1.79	709N	.39
318CN	1.49	710N	.79
319CN	1.29	711H	.39
320K-5	1.19	711N	.39
320K-12	1.19	723H	.55
320T-5	1.19	723N	.55
320T-12	1.19	725H	3.49
340K-5	1.19	733H	1.49
340K-12	1.19	733N	.99
340T-5	1.19	739N	1.19
340T-12	1.19	741N	.09
		747N	.79
		748H	.39
		748N	.39
		1414N	1.75
		1458	.69

UART

AY 5-1013A $4.98

factory prime

2114/4045 1K by 4 STATIC MEMORY

450nS. $11.95 650nS. $9.95

FROM ATARI

COLOR TELEVISION R.F. MODULATOR

$13.95

The Atari R.F. Modulator allows computer data to be displayed directly upon your existing television system. This unit converts the signal from the Apple II and other video sources into television frequencies. Operates from single 5 volt supply. Complete with metal case, mating R.F. connector and 15 feet of coax cable. Schematics and instructions included.

SPEAKERS

$.69 6 for $3.00

2¼ Miniature

Perfect for TV games, alarm systems, speech synthesizer or intercom. 8 ohms

5" HiFidelity $2.98

Thumbwheel switch

Ten position BCD

$1.39 ea.

10	50
1.19	.89

MINIATURE SWITCHES

your choice

$.98

10	50	100	1k
$.88	.81	.73	.66

SPDT Miniature Toggles

7101 C&K	ON-NONE-ON
7107 jbt	ON-OFF(mnt.ON)
7108 CK	ON-(moment.ON)
Rocker JBT	DPDT
Rotary	3P-4-Pos.
Rotary	3P-6-Pos.
Push B (N.O.)	$.39ea. 3/$1

TRIMMER POTENTIOMETERS

2K 5K 10K 50K

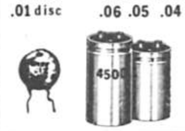
5 for $.98

20	50	100
16¢	14¢	12¢

CAPACITORS

ELECTROLYTICS

	ea.	10	50
80,000/10v.	3.95	3.49	2.95
4500/50v.	$1.49	1.35	1.19
1000/15v axial	$.55	.49	.45
.1 disc	$.12	.09	.07
.01 disc	.06	.05	.04



DIP Switch

$1.75 ea.

10	25	100
$1.63	1.52	1.29

specify 7 or 8 pos.

SPST

30 Conductor RIBBON WIRE

$.69 Ft.

SPECTRA-STRIP

Power Adapter

6 vdc, 140mA	$1.39
7 vdc, 1.4 A.	5.50
9 vdc, 15mA.	1.19
10 vAc, 300mA.	1.95
12 vdc, 600mA	2.95

RELAYS

SPDT MINIATURE

$1.39 ea.

10	25	100
$1.15	1.04	.89

Coil 12 Volt dc.

7 Amp Contacts

P.C. Board Mount

Transistors

	ea.	10	50	100
2N2222A	.20	.18	.16	.15
2N3055	.69	.65	.59	.55
MJ3055	.79	.75	.69	.65
2N3772	1.59	1.49	1.39	1.29
2N3904	.15	.11	.09	.07
2N3906	.15	.11	.09	.07

Diodes

		10	25	100
1N4002	100v.	.08	06	.05
1N4005	600v.	.10	.08	.07
1N4148	signal	.07	.05	.04

jumbo red LED's

ea.	10	25	100
$.15	.13	.11	.09

BINDING POSTS 5-WAY

3 for $1.19

20	100
$.35	.29

9 foot

$1.49

Heavy duty grounded power cord and mating chassis connectors.

SOLDERLESS TERMINALS

INSULATED

20 for $.98

Specify: 22-18; 16-14

100	500	1k
4.50	20.	35.

PANASONIC

$.98

Nicad

500mA "AA"

DISCOUNT Wire Wrap Center

IC SOCKETS

pin	wire wrap ea.	25	50	low profile ea.	25	50
8				17¢	16	15
14	37¢	36	35	18	17	16
16	38	37	36	19	18	17
24	99	93	85	36	35	34
40	169	155	139	63	60	58

50ft. $.98

KYNAR WIRE WRAP

500	1,000	11,000
$9.	$15.	$105.

Battery wire wrapping tool

$34.95 $29.95

OK HOBBY WRAP-30 wire wrap & strip tool

$5.45

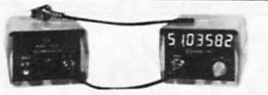
(213) 772-0800

VISA

master charge THE INTERBANK CARD

All merchandise sold by California Industrial is premium grade.
Orders are shipped the same day received.
PLEASE INCLUDE $1.00 SHIPPING ON ORDERS UNDER $15.00
California residents add 6% sales tax • Money back guarantee.
Sorry, no COD's • Foreign orders add 10%

MS-15 MINISCOPE $289.00 MS-15 MINISCOPE



SC-5 With Rechargeable Batteries & Charger Unit **$89**

FM-7 With Rechargeable Batteries & Charger Unit **$195**

NLS

Features Include: • By using the new NLS SC-5 Prescaler, the range of the FM-7 Frequency Meter, which is 10 Hz to 60 MHz, may be extended to 512 MHz (the upper VHF & UHF frequency bands). • The FM-7 utilizes an LED readout, providing 7-digit resolution. • The FM-7 can be calibrated to an accuracy of 0.00001%. • The SC-5 is accurate to one part per million. • Each unit has 30 millivolts sensitivity, is battery powered and has a charger unit included. • Dimensions of each are 1.9" H x 2.7" W x 3.9" D. • The units may be obtained separately or as a "Frequency Duo." • Parts & Labor guaranteed 1 year.

Tilt stand option .... $ 3.50
Leather case $16.00

MS-15 MINISCOPE $289

With Rechargeable Batteries & Charger Unit

- 15 megahertz bandwidth.
- External and internal trigger.
- Time base — .1 microsec. to 0.5 Sec/div - 21 settings. ± 3%.
- Battery or line operation.
- Automatic & line sync modes.
- Power consumption < 15 watts.
- Vertical Gain — .01 to 50 V/div - 12 settings ± 3%.
- Viewing area 1.1" x 1.35".
- Case size 2.7"H x 6.4"W x 7.5"D, 3 pounds.
- Parts & Labor guaranteed 1 year
- 10 to 1, 10 meg probe $24.50
- Leather carrying case $30.00

PROBE 1¢ with the purchase of SCOPE and the MENTION of this MAGAZINE

NEW VOLKSMETERS!

With LCD Display—
Excellent Readability in Direct Sunlight!

1% LM-300 Full three digits $99.50*

0.5% LM-350 3½ digits $125.00*

FEATURES

- Measures DC volts, AC volts, ohms and current.
- Automatic polarity, decimal and overload indication.
- No zero adjustment and no full scale ohms adjust.
- Large LCD display for easy reading without interpolation.
- Size: 1.9" H x 2.7" W x 4.0" D.
- Parts and labor guaranteed for one year.

SPECIFICATIONS

MODE	RANGE	ACCURACY LM-300	ACCURACY LM-350	RESOLUTION	INPUT RESISTANCE	TEST CURRENT
VOLTS DC*	1 10 100 1000	±1% Reading†	±0.5% Reading†	1 mV 10 mV 100 mV 1 V	10 MΩ	
VOLTS AC*	1 10 100 1000	±2% Reading† 50-400 Hz	±0.75% Reading† 50-400 Hz	1 mV 10 mV 100 mV 1 V	10 MΩ, 20 pF	
KILOHMS**	1 10 100 1000 10000	±1% Reading†	±0.5% Reading†	1 Ω 10 Ω 100 Ω 1 kΩ 10 kΩ		1 mA 100 µA 10 µA 1 µA 100 nA
CURRENT	1 mA 10 mA 100 mA 1 A	±2% Reading†	±2% Reading†	1 µA 10 µA 100 µA 1 mA	1 kΩ 100 Ω 10 Ω 1 Ω	

* 1000 vdc or peak ac maximum any range. †±2 Digits
**Test Voltage (Ohms) 1 vdc Full Scale - LM-300; 2 vdc Full Scale - LM-350
\# LM-350 has 100% over-range - full scale readings are 1.999, 19.99, etc.

Tilt Stand Option, add $3.50
Leather Case $16.00
3AA NiCad Batts. & Charger $12.00

Standard AA-size batteries provide up to 20 hours of operation.
Rechargeable NiCad batteries and charger unit available as optional equipment.
*Batteries not included.

LEDU MG 10A List $72.00

Perfectly balanced, fluorescent lighting with precision magnifier lens. For prof'l, techn'l & hobbist. Has die cast protective shade, inst. start 3 diopter lens. 42" reach.

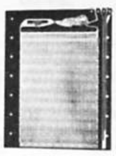
$44^{95}

with T-9 fluorescent lamp (required) $49.95
colors: Gray or Black Chocolate Brown
2 choices please

$44^{95} SPECIAL

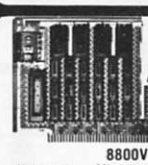
8803 MOTHER BOARD FOR S100 BUS MICRO-COMPUTERS — Vector

- Kit includes 12 tantalum capacitors for +5, +12, —12 buses and insulated mounting spacers.
- Wiring side shown. Component side bare epoxy glass with white markings for component locations.
- G10 epoxy glass board with 2 ounce copper, solder plated and .038 diameter holes for leads.
- Solder mask with solder windows on etched circuits to avoid accidental short circuits.
- Mounts 11 receptacles with 100 contacts (2 rows) on .125 centers with .250 row spacing. Vector part number R681-2, or mounts 10 receptacles plus interconnections to smaller mother board for expansion.
- Includes etched circuits and instructions for option of active, pull-up, or floating terminations.
- Large buses: +5V and GND (10 AMPS), ± 12V or 16V (7 AMPS). Current ratings are per MIL-STD-275 with 10°C rise.
- Fits in Vector-pak enclosures.
- Fits in IMSAI 8080 microcomputer as expander board.



Price: $29.50

Vector Plugboards



8800V

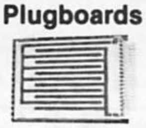
Universal Microcomputer/processor plugboard, use with S-100 bus. Complete with heat sink & hardware. 5.3" x 10" x 1/16"

1-4	5-9	10-24
$19.95	$17.95	$15.96

8801-1

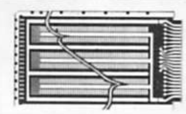
Same as 8800V except plain; less power buses & heat sink.

1-4	5-9	10-24
$14.95	$13.46	$11.96



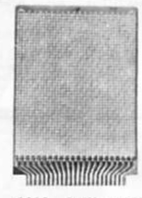
3682 9.6" x 4.5" $10.97
3682-2 6.5" x 4.5" $9.81

Hi-Density Dual-In-Line Plugboard for Wire Wrap with Power & Grd. Bus Epoxy Glass 1/16" 44 pin con. spaced .156



3677 9.6" x 4.5" **$10.90**
3677-2 6.5" x 4.5" **$9.74**

Gen. Purpose D.I.P. Boards with Bus Pattern for Solder or Wire Wrap. Epoxy Glass 1/16" 44 pin con. spaced .156



3662 6.5" x 4.5" **$7.65**
3662-2 9.6" x 4.5" **$11.45**

P pattern plugboards for IC's Epoxy Glass 1/16" 44 pin con. spaced .156

R644 P.C. recepticle 22/44 cont. .156 ctrs. pierced solder eyelets tails gold $3.45
R644-3 P.C. recepticle 22/44 cont. .156 ctrs. Wire Wrap tails gold $4.49

S-100 Bus P.C. Edge Connectors

R681 DIP solder tails on .140" spaced rows for ALTAIR mother boards. Fits .042" dia. holes, gold $7.35
R681-1 .025" sq., 3 wrap lengths posts (.62 long) on .250 spaced rows, gold $5.00
R681-2 .025" sq., 1 wrap length posts (.22 long) on .250 spaced rows for wrapping or DIP solder for IMSAI mother boards, gold $5.85
R681-3 pierced solder eyelet tails, gold $7.35

1/16 Vector BOARD

.042 dia holes on 0.1 spacing for IC's

Phenolic

PART NO.	SIZE	PRICE
64P44-062XXP	4.5 x 6.5"	$ 1.49
169P44-062XXP	4.5 x 17"	$ 3.51

Epoxy Glass

PART NO.	SIZE	PRICE
64P44-062	4.5 x 6.5"	$ 1.70
84P44-062	4.5 x 8.5"	$ 2.10
169P44-062	4.5 x 17"	$ 4.30
169P84-062	8.5 x 17"	$ 6.39

SLIT-N-WRAP

Wraps insulated wire on .025" square posts
FOUR TIMES FASTER than regular manual wrap-post tools

P180 with two 100' spools of 28 ga. wire **$24.50**

P160-4T includes charger, wire **$75.00**

Vector

NO PRE-STRIPPING*
NO PRE-CUTTING*
SPOOL-FED WIRE*

*The spooled wire passes through the tool past a slitting edge next to the wrap post. A narrow longitudinal cut is made in the insulation where it presses the square post corner; the bared copper is indented by the sharp edge. (7 turns = 28 contacts) Insulation is slit where wrapped but not between terminal posts when led straight out of the tool.

"Daisy-chain" runs
Reliable and easy to use
Manual or power operation

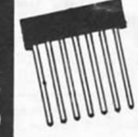
SLIT-N-WRAP WIRE $2^{75} Package
NO. 28 GAGE INSULATED WIRE, 100' SPOOLS
W28-2-A-Pkg. 3. Green W28-2-C-Pkg. 3. Clear
W28-2-B-Pkg. 3. Red W28-2-D-Pkg. 3. Blue

2708 8K 450 ns EPROM FACTORY PRIME

1-9	$12.00
10-24	$11.50

25 + *Call for Price*

14 & 16 PIN 3 LEVEL WIRE WRAP SOCKETS



14-T3 100 for $30.00
16-T3 100 for $30.00
50 of ea. for $32.00

ACE • All-Circuit Evaluator — for fast, solderless, plug-in circuit building and testing

Model 236

10% OFF

Just plug in any components with leads to .032" dia. Interconnect with solid wire up to 20 ga. Assembled models too!

ACE Model No.	Tie Points	DIP Capacity	No. Buses	No. Posts	Board Size (inches)	Price Each
200-K (kit)	728	8 (16's)	2	2	4-9/16 x 5-9/16	$18.95
208 (assem.)	872	8 (16's)	8	2	4-9/16 x 5-9/16	28.95
201-K (kit)	1032	12 (14's)	2	2	16 x 7	24.95
212 (assem.)	1224	12 (14's)	8	2	4-9/16 x 7	34.95
218 (assem.)	1760	18 (14's)	10	2	6-1/2 x 7-1/8	46.95
227 (assem.)	2712	27 (14's)	28	4	8 x 9-1/4	59.95
236 (assem.)	3648	36 (14's)	36	4	10-1/4 x 9-1/4	79.95

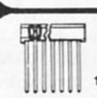
Gold anodized aluminum base/ground; non-corrosive nickel-silver terminals; 4 rubber feet

SALE

LIQUID CRYSTAL DIGITAL CLOCK-CALENDAR

- For Auto, Home, Office
- Small in size (2x2½x½)
- Push button for seconds release for date.
- Clocks mount anywhere with either 3M double-sided tape or VELCRO, included.
- 2 MODELS AVAILABLE:
  LCD-101, portable model runs on self-contained batteries for better than a year.
  LCD-102, runs on 12 Volt system and is back-lighted.
- LCD-101 or LCD-102 your choice .... $34.95 ea
- Clear desk stand for $2.00

3 LEVEL GOLD WIRE WRAP SOCKETS



	1-24	25-49	50-99
8 pin	.47	.42	.37
10 pin	.45	.41	.37
14 pin	.39	.38	.37
16 pin	.43	.42	.40
18 pin	.75	.68	.62
22 pin	1.00	.97	.95
24 pin	1.00	.94	.75
28 pin	1.09	.98	.84
36 pin	1.59	1.45	1.30
40 pin	1.49	1.39	1.29

Vector MICRO-KLIP

for .042 dia. holes (all boards on this page)
T42-1 pkg. 100 **$ 1.50**
T42-1 pkg. 1000 .... **$11.00**
P-149 hand installing tool .... **$ 2.03**

.8" LED ALARM CLOCK

12 hr. LED Alarm Clock uses 3½ digit .8" LED Display with AM/PM indicators and colons. Direct drive, PIN to PIN interface with **S1998A** I.C. Just add switches, AC Supply, Alarm. Display and I.C. only.

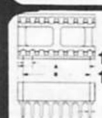
$7.95 or 2/$15.00

Price Breakthrough! $17^{95}

MA1003 CAR CLOCK

Bright Green Fluorescent Display Crystal Time Base Assembled, just add switches and 12 VDC.

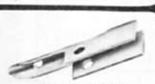
SPECIAL



14CS2 100 for $14^{00}
16CS2 100 for $16^{00}
14 pin CS2 10 for $2^{00}
16 pin CS2 8 for $2^{00}

These low cost DIP sockets will accept both standard width plugs and chips.
For use with chips, the sockets offer a low profile height of only .125" above the board. These sockets are end stackable.

WRAP POST



for .042 dia. holes (all boards on this page)
T-44 pkg. 100 .. **$ 2.28**
T-44 pkg. 1000 .... **$14.00**
A-13 hand installing tool .... **$ 2.80**

PRIORITY ONE ELECTRONICS

P21

10031 W Woodley Ave. Sepulveda CA. 91343

VISA master charge

Terms: VISA, MC, BAC, check, Money Order, C.O.D., U.S. Funds Only. CA residents add 6% sales tax. Minimum order $10.00. Orders less than $75.00 include 10% shipping and handling; excess refunded. Just in case . . . please include your phone no. **"Sorry, no over the counter sales"** Good thru June 15, 1978

Send for our latest brochure. **phone orders welcome (213) 893-8202** OEM and Institutional inquiries invited.

24 PIN DIP PLUGS WITH COVERS

3 / $1.00
40 / $10.00

NLS MS-15 MINISCOPE $289.00 NLS MS-15 MINISCOPE $289.00

16K E-PROM CARD

New!

Uses 2708's!

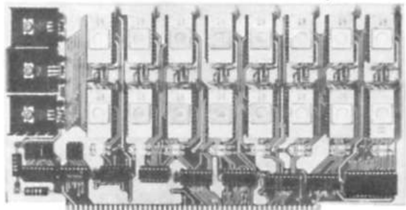
IMAGINE HAVING 16K OF SOFTWARE ON LINE AT ALL TIME!

S-100 (Imsai/Altair) Buss Compatible!

KIT FEATURES:

1. Double sided PC board with solder mask and silk screen and gold plated contact fingers.
2. Selectable wait states.
3. All address lines & data lines buffered!
4. All sockets included.
5. On card regulators.

KIT INCLUDES ALL PARTS AND SOCKETS (except 2708's). Add $25. for assembled and tested.



DEALER INQUIRIES INVITED!

$57.50 kit

SPECIAL OFFER:

Our 2708's (450NS) are $12.95 when purchased with above kit.

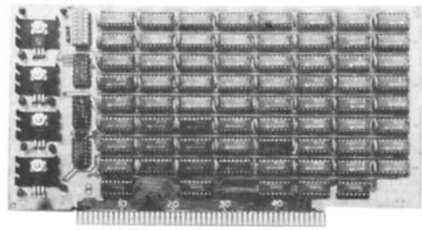
Fully Static!

8K LOW POWER RAM KIT-$149.00

S-100 (Imsai/Altair) Buss Compatible!

KIT FEATURES:

1. Doubled sided PC Board with solder mask and silk screen layout. Gold plated contact fingers.
2. All sockets included.
3. Fully buffered on all address and data lines.
4. Phantom is jumper selectable to pin 67.
5. FOUR 7805 regulators are provided on card.



USES 21L02 RAM'S!

Fully Assembled & Burned In
$179.00
Blank PC Board w/ Documentation
$29.95
Low Profile Socket Set 13.50
Support IC's (TTL & Regulators)
$9.75
Bypass CAP's (Disc & Tantalums)
$4.50

MOTOROLA QUAD OP - AMP
MC 3401. PIN FOR PIN SUB. FOR POPULAR LM 3900.
3 FOR $1

RECTIFIER SPECIAL
1.5 AMP 100PIV
EPOXY CASE AXIAL LEADS
15 FOR $1

FULL WAVE BRIDGE
4 AMP. 200 PIV.
69¢ EA. 10 FOR $5.75

NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE SUPPLIERS OF CPM SOFTWARE.

MOTOROLA 7805R VOLTAGE REGULATOR
Same as standard 7805 except 750 MA output. TO–220. 5VDC output.
44c each or 10 for $3.95

450 NS! 2708 EPROMS 450NS!
Now full speed! Prime new units from a major U.S. Mfg. 450 N.S. Access time. 1K x 8. Equiv. to 4-1702 A's in one package.
$15.75 ea.
Special Offer: $12.95 each when purchased with our 16K EPROM kit!

Z–80 PROGRAMMING MANUAL
By Mostek, The major Z-80 second source. The most detailed explanation ever on the working of the Z-80 CPU CHIPS. At least one full page on each of the 158 Z-80 instructions. A MUST reference manual for any user of the Z-80. 300 pages. Just off the press! A D.R.C. exclusive! $12.95

CMOS SPECIAL!

CD 4001 – 5 for $1.	CD4040 – $1. each
CD 4011 – 5 for $1.	CD4042 – 2 for $1.
CD 4013 – 3 for $1.	CD 4049 – 3 for $1.

POWER RECTIFIER #2 Motorola
Stud Mount. 1N1187.
35 AMPS. 300 PIV.
Military Quality! $1.19 ea. or 4/$3.50

3 AMP RECTIFIER
1N4721. Axial Lead. 200 PIV.
HEAVY DUTY!
House numbered

4 for $1.00 Metal Case

4K STATIC RAM'S
2114. The new industry standard. Arranged as 1K x4. Equivalent to 4-21 L02's in 1 package! 18 pin DIP. 2 chips give 1Kx8.
2/$24. 8/$85.

OPCOA LED READOUT
SLA-1. Common Anode. .33 inch character size. The original high efficiency LED display. 75c ea. or 4 for $2.50

EXPERIMENTER'S CRYSTAL
262.144KHZ. This frequency is 2 to the 18th power. Easily divided down to any power of 2, and even to 1HZ. New by CTS-Knight. A $5. value!
$1.25 each

GE Ni-Cad Battery Pack
3 Cell pack, gives 4 volts at 900MAH. Brand new, factory fresh. Each cell is 2/3 "C" size. **$4.50 /pack. Buy 3 packs (12 volts) for $10.95. Limited stock!**

741C OP AMPS
Mini Dip. Prime new units. Has computer Mfg's house number.
12/$2. 100/$15.

Disc Capacitors
.1 MFD 16V. P.C. leads. Most popular value. By Sprague.
20/$1.00

HEAVY DUTY!
Full Wave Bridge
25AMP 50PIV
$1.25

Tantalum Capacitors
1 MFD. .35V. By Kemet. Axial Lead. Best Value! 10/$1.

2N3904-House No.
TO–92. NPN.
VCEO-45.
HFE 100 to 300
10 for $1.00

Jumbo Red Leds
New by G.E. Like MV5024. Number SSL-22.
6/$1. 25/$3.75

12VDC Relay by CDE. SPDT. Coil is 280 ohm. Small Size. $1.29

GE 10 AMP Triac
SC146D. House no. To-220 case. Rated 10 amps 400PIV.
75cea. 3/$2.

LS SERIES TTL

74LS00 — 33c	74LS74 — 49c
74LS02 — 35c	74LS90 — 69c
74LS04 — 35c	74LS138—89c
74LS08 — 35c	74LS154—1.49
74LS10 — 33c	74LS175—1.10
74LS20 — 33c	74LS367—75c
74LS73 — 49c	74LS368—85c

D20

Digital Research Corporation
(OF TEXAS)

P. O. BOX 401247 • GARLAND, TEXAS 75040 • (214) 271-2461

TERMS: Orders under $15. add 75c. No COD's. We accept VISA, MasterCharge and American Express Cards. Money Back Guarantee on all items! Texas Residents add 5% Sales Tax. WE PAY POSTAGE!

ADVANCED COMPUTER PRODUCTS

THE FIRST TO OFFER PRIME PRODUCTS TO THE HOBBYIST AT FAIR PRICES NOW LOWERS PRICES EVEN FURTHER! A39

1. Proven Quality Factory tested products only, no re-tests or fallouts. Guaranteed money back. We stand behind our products.

2. Same Day Shipment All prepaid orders with cashiers check, money order or charge card will be shipped same day as received.

MOTOROLA 6800 COMPATIBLE MODULES

MEK 6800 D2 KIT	$235.00
9600 6800 MPU	595.00
9601 16 Slot Mother Bd.	175.00
9602 16 Slot Card Cage	75.00
9603 8 Slot Mother Bd.	100.00
9610 Proto Board	36.00
9615 4K EPROM Module (1702A)	250.00
9620 16 Port Parallel I/O	375.00
9626 8K Static RAM Module	295.00
9626K 8K Static RAM Kit	225.00
9630 Extender Card	60.00
9650 8 Port Duplex Asyn Serial I/O	395.00
Connectors $6.50	6/84.00

All assembled & tested not Kits

PLUS MOTOROLA TV MONITORS-PRIME

Model M3560-155 L01 12" display	219.95
Model M2000-155 9" display	199.95

Add $10.00 for shipping

LOGOS 8K STATIC MEMORY

$125.99 Kit

Assembled & tested $179.95
250 ns. Kit 149.95
Assembled & tested 199.95

NOW 1 Year Field Experience

Features: Lowpower, Dip Switch Selectable memory protect down to 256 Bytes, addressing on 1K Boundaries, No wait states, fully buffered, battery back-up.

SPECIAL OFFER:
Buy 4 Units Only $117.00 ea.

BYTEUSER 8K EPROM $64.95

Assembled & tested $ 94.95
Bare PC Board w/Data 21.95
8K EPROM (8) 2708 $112.00

Features: Power on Jump, Reset Jump, all socketed with top quality PC Board material.

SPECIAL OFFER:
Buy 4 Units Only $59.95 ea.

Z-80 CPU KIT $129.95 Kit

Low Price
(regular price 269.95)

Assembled & tested $199.95

FEATURES:
S-100 IMSAI/Altair compatible, completely compatible to TDL hardware and software. Can be used at 4MHz with Z-80A.
Add $7.00 for sockets.
Add $5.00 for Z80 manual.

NOW IN STOCK
FOR IMMEDIATE SHIPMENT.

DATA BOOKS

NSC Digital	$ 3.95
NSC Linear	4.95
NSC Linear A/N Vol. I	2.95
NSC Linear A/N Vol. II	2.95
NSC CMOS	2.95
NSC Audio	2.95
NSC Voltage Reg	2.95
NSC Memory	3.95
Intel Data Book	3.95
Intel MCS 85 Manual	4.95
Intel MCS 80 Manual	7.95
Intel MCS 40 Manual	4.95
AMD 8080 Microproc. Handbook	7.95
AMI MOS Catalogue	3.95
Raytheon Linear	2.50
Raytheon Quads/Duals	1.95
GI MOS Catalogue	4.95
Osborne Intro to Micro Vol 0	7.50
Osborne Intro to Micro Vol I	7.50
Osborne Intro to Micro Vol II	15.00
Osborne 8080 Programming	7.50
Osborne 6800 Programming	7.50
Osborne Z80 Programming	7.50

AMI 6800 KIT EVK 99

only $133.00

Send for complete details

EVK 99 Kit	$133.00
EVK 100 Kit	269.95
EVK 200 Kit	449.95
EVK 300 Assembled	699.00
Universal Kluge Board	95.00
16K Byte RAM Board	75.00
6 Slot Motherboard	35.00
Extender Board	45.00
Video Board (avail. future)	95.00
Connectors . . $6.50	12/59.95
Solid Frame Chassis	120.00
Frame Chassis	69.95
Micro Assembler ROM	30.00
Proto ROM	30.00
Tiny Basic Papertape	20.00
Tiny Basic EPROM	125.00

Hardware or Programming Manuals 15.00 ea.

8080A CPU KIT $99.95 Kit

Assembled & tested $129.95

FEATURES:
S-100 bus compatible, complete CPU with eight level vector interrupt. Includes sockets.
Add $5.00 for 8080 Manual

16K SUPERFAST RAM'S

µp D 416/4116 350ns. PRIME

Buy 8 pcs. – Only $29.95 each
Buy 16 pcs. – Only $27.95 each

Larger quantity or group buys call for current price quote.

TARBELL FLOPPY INTERFACE

Complete Kit Only $179.95
Assembled & tested $269.95

FEATURES: S100/Altair/IMSAI compatible, compatible to most disc drives including Persci, Innovex, GSI, Shugart and others.

SOFTWARE: Uses CPM which is available for $70.00
CPM documentation add $20.00

Same Day Shipment

NAKED PC BOARD SALE S-100

Z-80 CPU	$31.95
2708 EPROM 8K	21.95
8K Static RAM	21.95
32K Static RAM	59.95
Floppy I/O	$39.95
Cassette I/O	29.95
Proto Bd.	27.95
Extender	15.95

UV "EPROM" ERASER

Model UVS-11E only $59.95

Holds (4) chips at a time. Special holding tray with UVA Absorber. Exclusive safety interlock system.

Backed by 45 years UV experience.

WOW 1771 FLOPPY SPECIAL

$32.95

While they last $32.95
regular price $55.95
Western digital P/N 1771A with App. Note & Data
Only 100 units available

MICROPROCESSORS

• Z-80	$21.95
Z-80A	29.95
F-8 (3850)	19.95
2650	24.95
CD1802	19.95
8080A	12.95
8085	29.95
8008-1	14.95
2901	21.95
2901A	29.95
TMS 9900JL	89.95
CP1600	39.95
6502	19.95
IM6100	29.95
• 6800P	19.95
6802P	32.95

SUPPORT DEVICES

• 3881 (Z80 PIO)	$12.95
• 3882 (Z80 CTC)	12.95
3851 (F8 Program)	14.95
3853 (F8 Memory I/O)	14.95
8212 8 bit I/O	3.50
8214 Priority Interrupt	9.95
8216 Bus driver	3.75
8224 Clock Gen	4.85
8224-4 4MHz CLK Gen	12.95
• 8T26 Busdriver	2.25
8226 Busdriver	3.95
8228 Sys Controller	7.95
8238 Sys Controller	9.95
8251/9551 Prog. Comm. I/O	11.95
8253 Interval Timer	21.95
8255/9555 Prog. Periph. I/O	11.95
8257 DMA Control	32.95
8259 Prog. Interrupt	32.95
6810-1 128 × 8 RAM	4.95
• 6820 PIA	7.95
6834-1 EPROM 4K	14.95
6834 EPROM 4K	16.95
6850 ACIA	9.95
6852 Serial Adapter	11.95
6860 Modem	14.95
6862 RPS Modulator	17.95
• 6871B 1.0 MHz Clock OSC	26.95
6880 MPU Busdriver	2.95
68 MinII Minibug 2	49.95
68 MinIII Minibug 3	74.95
6831/10224 Micro Assembler Rom	30.00
6831/11003 Proto Rom	30.00
1821SCD 1K RAM	15.95
1822SCD 256 × 4 RAM	16.95
1824CD 32 × 8 RAM	9.95
1852CD 8 bit I/O	10.95
1856CD I/O	8.95
1857CD I/O	8.95

PROM'S

• 2708	$14.95
2708S 650ns	9.75
1702A	7.95
• 1702-6	3.50
5204	16.95
• IM5610	2.95
6834	16.95
6834-1	14.95
82S123	2.95
82S126	4.50
82S129	2.95
8223	2.95
5203	6.95

Programming Available

RAMS DYNAMIC

• 4115	$18.95
4116 16K	34.95
TMS4050	4.95
TMS4060 22 pin	4.25
WD4060 22 pin	3.25
4096 16 pin	4.95
2104	4.95
2107	3.95
• 2107B-4	4.25
MM5261	3.95
MM5262	1.95
MM5270	4.95
MM5280	4.95
1103	1.95
4008 L	2.95

FLOPPY DISK

WD 1771 Floppy	32.95
WD 1771 H01	52.95

CHARACTER GENERATORS

R032513 Upper 5V	$10.95
R032513 Lower 5V	9.95
HD0165	9.95
MC6571A	10.80
MC6574	14.50
MC6575	14.50

KEYBOARD ENCODERS

AY52376	$14.95
AY53600	14.95

UARTS/USRTS

AY51013A (5V)	$ 5.50
TR1602B (5V)	4.50
AY51014 (5-14V)	8.95
AY51015 (5V)	9.95
IM6402 (5V)	12.95
IM6403 (5V)	12.95
S2350	10.95
WD1671B Astros	19.95

BAUD RATE GENERATORS

MC14411	$11.95
WD1941 Dual	9.95
34702	13.95

TV CHIPS

• MM5320 TV Synch	$ 9.95
MM5369 Prescaler	3.95
MM57100 Game Chip	16.95
MM57104 Clock	3.75
LM1889 Modulator	3.95
CW300 Saw Function	1.50
AY38500-1 TV Game	9.95
AY38600-1 Color TV Game	24.95
AY38615-1 Color Converter	8.95
• AY38700-1 Tank Chip	29.95
RF Modulator	8.95

ORGAN CHIPS

MM5554
MM5555 (1 each) NSC Set $19.95
MM5556

CLOCK CHIPS

MM5314	$ 4.95
MM5316	4.95
MM5375	4.95

STATIC RAM HEADQUARTERS

We maintain largest inventory of Static Rams in the West

	1-24	25-99	100		1-24	25-99	100
				2114(450 ns)	14.95	14.25	13.45
				4200A	12.95	12.75	11.45
• 21L02(450 ns)	1.30	1.25	1.15	• upD410 (4200)	10.95	10.50	9.95
21L02(250 ns)	1.95	1.80	1.50	4402	12.95	12.75	11.45
2102	1.25	1.15	1.10	5101C-E	11.95	11.25	10.25
2102-1	1.30	1.25	1.15	74C89	3.25	3.05	2.85
91L02APC	1.75	1.65	1.50	7489	2.25	1.99	1.88
21L11-1	4.25	4.10	3.95	8599	1.88	1.75	1.62
2111-1	4.10	3.95	3.85	74S189	4.50	4.30	4.25
2112-1	3.00	2.80	3.69	74S200	4.50	4.30	4.25
21L01-1	2.95	2.85	2.75	74S201	4.50	4.30	4.25
2101-1	2.95	2.85	2.75	1101A	1.49	1.29	1.10
2114(650 ns)	12.95	12.25	11.45	3107	3.95	3.70	3.25

A TO D CONVERTORS

8700CJ	$13.95
8701CN	21.95
8750CJ	13.95
1408L8	5.95
• 148L6	3.95

DISPLAYS/OPTO

DL 704/707 CC/CA .300	1.25
FND359 CC .357	.95
FND 500/507 CC/CA .500	1.35
FND 503/510 CC/CA .500	.95
FND 800/807 CC/CA .800	2.50
Bowmar 9 digit bubble	.99
FSC 8024 4 digit CC .800	4.95
HP7340 HEX Display	15.95
TIL 305 5 × 7 Array	4.50
TIL 306 7 seg w/logic	8.95
TIL 308 7 seg w/logic	8.95
TIL 309 7 seg w/logic	7.95
TIL 311 HEX Display	9.25
MA1003 12 auto clock	17.95
MA1002 4 digit clock module	9.95
MA1010 4 digit clock module	9.95
NSN 373/374 dual CC/CA .300	2.20
NSN 583/584 dual CC/CA .500	2.60
NSN 783/784 dual CC/CA .700	3.00
4N25 Opto Isolater	3/1.99
MCT 2 Opto Isolater	.89
4N33 Darlington ISOL	1.75
Red Led's .185 Dia	5/1.00
Green/Yellow	4/1.00

SOCKETS

8 Pin ww	.32	8 Pin S/T	.17
14 Pin ww	.37	14 Pin S/T	.20
16 Pin ww	.38	16 Pin S/T	.22
18 Pin ww	.60	18 Pin S/T	.31
20 ww	.90	22 Pin S/T	.35
22 Pin ww	.93	24 Pin S/T	.41
24 Pin ww	.93	28 Pin S/T	.49
40 Pin ww	1.45	40 Pin S/T	.63

CONNECTORS

6 Pin Single S/E	1.49
15/30 Dual S/E	1.95
18/36 Dual S/E	2.35
22/44 Dual S/E	2.95
43/86 Dual S/T	6.50
43/86 Dual W/W	6.50
50/100 IMSAI W/W	4.75
50/100 IMSAI S/T	4.75
50/100 Altair W/W	5.95
50/100 Actair S/T	5.95

25 Pin D Subminature

DB25P	3.25
DB25S	3.75
Hood	1.25

IMSAI Card Guides 4/1.00

RS232 CONNECTOR SET W/Hood $6.50

CRYSTALS

1.0MH$_3$	$ 5.85	10.0MH$_3$	4.95
2.0	5.85	13.0	4.95
2.097152	5.85	14.31818	4.95
2.4576	5.85	18.0	4.95
3.579545	3.95	18.432	5.95
4.0	4.95	20.0	4.95
4.194304	5.95	22.1184	5.95
4.91520	5.95	27.000	5.95
5.0	4.95	36.000	5.95
5.7143	5.95	48.000	5.95
6.0	4.95	100KC	12.95

NEW CTS DIPSWITCHES

CTS206-4	$1.75	CTS208-8	$1.95
CTS206-5	$1.75	CTS209-9	$1.95
CTS206-6	$1.75	CTS209-10	$1.95
CTS207-7	$1.75		

We also stock full of 7400, 74LS, 74L Linear and CMOS. Send for pricing or use our competitors STD catalog pricing.

NEW
1978 CATALOGUE
AVAILABLE FEB.
SEND .25 POSTAGE

74LS00 TTL

74LS00	.22	LS86	.55
LS02	.24	LS90	1.10
LS04	.24	LS92	1.10
LS05	.28	LS93	1.10
LS08	.28	LS95	1.69
LS10	.28	LS107	.56
LS13	.79	LS109	.56
LS14	1.69	LS112	.56
LS20	.26	LS113	.56
LS27	.38	LS123	.99
LS30	.26	LS132	1.10
LS32	.34	LS136	.99
LS40	.26	LS138	1.39
LS42	1.10	LS139	1.49
LS51	.26	LS151	1.49
LS73	.56	LS153	1.29
LS74	.56	LS155	1.49
LS75	.79	LS157	1.29
LS76	.59	LS158	1.29
LS83	1.69	LS161	1.39
LS85	1.99	LS162	1.39

LS163	1.39	LS367	.99
LS164	1.39	LS368	.99
LS174	1.39	LS377	1.29
LS175	1.39	74200	5.96
LS190	2.49	74251	1.69
LS191	2.49	74279	1.10
LS192	2.49	74298	1.99
LS193	2.49	74365	.79
LS194	2.49	74366	.79
LS195	.52	74367	.79
LS251	1.39	74368	.79
LS253	1.49	LS378	1.29
LS257	1.29	81LS95	1.10
LS258	1.29	81LS96	1.10
LS279	1.10	81LS97	1.10
LS283	1.10	81LS98	1.10

MONTHLY SPECIALS

up D371 Mag Tape Control	$49.95
AY5-3550 4 3/4 Digit DVM	24.95
MK5007/5009 Counter	6.95
AY5-3507 DVM Chip	12.95
ICM7208IPI Ctr/Disp/Driv	16.95
ICM7209 IPI Ctr/Driver	19.95
ICM7045 IPI Stop Watch	18.95
G115M 6 Channel MOS FET SW	1.50
4116/416 16K Dynamic RAM	34.95
Intel 3404 6 bit Latch	2.95
1488/1489 RS232 Driver	1.25
8T97 Buffer	1.25
8130/8131	2.45
8833/8835	1.99
74367/74368	6/5.00
75451/52/53	10/2.50
8T26	2/4.50
78L05	2/1.00
LM318H	1.00

4N25 Opto/Iso .69

SPECIAL 7/$1.00

7405 MPS6530
7416 MPS3568
7438 MPS6516
7440 MPS6522

SPECIAL 2/$1.00

7442 74123
7490 74145
7493 74153
7495 74LS195

MCT2	$ .89	MPS3642	5/1.00
2N3638A	5/1.00	MPS6571	5/1.00
2N3640	5/1.00	MPS3693	5/1.00
2N2369A	5/1.00	MPS6530	5/1.00
2N3692	5/1.00	MPS3646	5/1.00
MSPS4318	5/1.00	7525J	.50
MPS5401	5/1.00	7524J	.60
MPS6516	5/1.00	74S258	.60
MPS6544	5/1.00	7489	1.49
MPS3568	5/1.00	74145	.60
MPS6522	5/1.00	74H55	.19
MPS404A	5/1.00	715DC	1.49
2N5462	5/1.00	74199	1.00

COMP KITS & SYSTEMS:

MEK6800D2 Kit	$235.00
KIM1 6502	245.00
EVK99 6800 Kit	133.00
Technico 9900 Kit	299.00
Intercept Jr. 6100 Kit	281.00
Iasis 8080 Computer Book	499.00
NSC SC/MP Kit	99.00
NSC Keyboard Kit	95.00
Low Cost S-100 BUS with 8 slots & Power Supply	159.95
8K Ram Kit (Logos I)	125.95
Byteuser 8K Eprom Kit	64.95
ZPU Kit (Z80)	295.00
TDL ZPU Kit (Z80)	269.00
Tarbell cassette I/O	115.00
S100 8 slot Motherboard with connectors (expandable)	69.95
S100 Extender Board	15.95
S100 Proto Board	27.95
Vector 8800 Proto Board	19.95
IMSAI 8080 w/22 slot	699.00
Cromemco Z-2	595.00
TDL XITAN alpha 1	769.00
Computalker	385.00
Heuristics Speechlab	299.00
SOROC IQ120 Terminal	995.00
8K Ram Board (Logos)	21.95
8K Eprom Board	21.95
S100 32K Ram Kit	875.00
Bytesaver Kit	145.00

NOTICE

We are looking for software support packages to offer in our new catalogue. Please write or call if you want to participate.

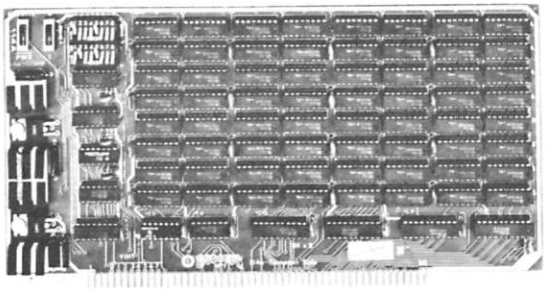
All Shipments FCM or UPS. Orders under $100.00 add 5% handling and postage. Orders over $100.00 add 2.5% handling & postage. Mastercharge Bank americard COD accepted w 25% deposit. California Residents add 6% tax. Foreign Orders add 8% handling. All parts prime factory tested guaranteed. Same day shipment. Add 25 cents for Data.

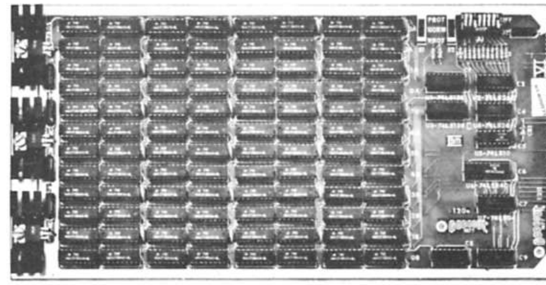
Retail pricing may vary from Mail Order Pricing.

P. O. BOX 17329 Irvine, California 92713 New Phone (714) 558-8813
TELEX/TWX: 910-595-1565

Retail Store Open Mon. – Sat.
Located at 1310 "B" E. Edinger,
Santa Ana, CA 92705

Beautiful Boards





OUR BEST SELLER: ECONORAM II™

S-100 Compatible 8K × 8 in a cost-effective package. Buffering on all lines, 0 wait states with the 8080, low power consumption, configured as two separate 4K blocks for addressing flexibility, handles DMA, memory protect with vector interrupt provision if you try to write into protected memory, fully socketed, gold-flashed edge fingers, solder masked and legended board . . . this is the board that doesn't cut any corners, but cuts the price instead.

(See the 1/77 issue of **Kilobaud** magazine for a product profile that tells just about everything you'd ever want to know about Econoram II . . . or send a self-addressed, stamped envelope to "Kilobaud Article" c/o our address and we'll send you a reprint. But if you **really** want to be convinced . . . talk to somebody who owns one!)

Kit form: $135.00 3 kits: $375.00 Assembled, tested: $155.00

SUPER MEMORY FOR A SUPER MACHINE: H8 COMPATIBLE ECONORAM VI™

Users of the S-100 buss have found out why our memories are their best value . . . now H8 owners can find out too. This 12K × 8 kit offers the same basic features as our ECONORAM series . . . static design, configuration as two blocks (one 8K and one 4K), switch selected protect, sockets for all ICs, full buffering on address and data lines . . . plus the required hardware and edge connector to mate mechanically with the H8. As a bonus, **all** sockets and bypass capacitors are pre-soldered to the circuit board so you can start right in on the fun part of building this high-quality memory.

Kit form: $235.00

ANNOUNCING . . . THE 16K ECONORAM IV™

We'll be ready to ship these soon, so we thought you'd like a sneak preview. The price? Under $400. The performance? All that you've come to expect from the Econoram line, along with impressively low power consumption and a couple of other tricks we have up our sleeve. If you've been waiting for a 16K board, you'll be happy you waited for us.

SOME WORDS ABOUT STATIC MEMORIES

When it comes to memory, we're pretty partial to static technology. Although more costly than dynamic devices, static memories are free of critical refresh and timing needs — which is one reason why DMA works so well with our memory boards. When we send an Econoram out into the world, we not only want it to work right with whatever system you have (Altair, IMSAI, Cromemco, Parasitic, Polymorphic, etc.); we want it to keep working for you. Static memories are proven, time-tested, and reliable . . . that's why we like them so much.

ACTIVE TERMINATOR BOARD

The active termination circuitry in our motherboard kits minimizes the ringing, crosstalk, overshoot, scrambled data, and noise problems that can occur with unterminated lines. But even if you don't have a Godbout motherboard, you can trick your computer into thinking you do by adding this useful peripheral. Simply plug into any S-100 machine, and gain the benefits of active circuitry. **#CK-017, $29.50.** Kit form only.

CPU POWER SUPPLY

Here is an economical supply for small computer systems or digital bench work. Delivers 5V @ 4A with crowbar overvoltage protection (accidents can happen . . . and you shouldn't have to replace all your TTL if one does!). Also gives +12V @ ½A and −12V @ ½A, along with an adjustable negative bias supply (−5 to −10V @ 10 mA). All in all, you can't beat the price or the performance. **#CK-014, $50.00.** Kit form only.

DB-25 RS-232 SUBMINI-D CONNECTORS

Male plug, **#CK-1004, $3.95;** female jack, **#CK-1005, $3.95;** plastic hood for male connector, **#CK-1006, $0.90.**

PLUG FROM BILL: There's more to life than computers . . . like music. Craig Anderton, noted author and designer of our Musikit products, has produced a cassette tape of original music that is distributed by our friends at PAIA Electronics (1020 W. Wilshire, Oklahoma City, OK 73116; $6.45 ppd). In addition to hearing our Musikits in action, you get to hear some really good modern music. We like it . . . you probably will too.

WE ALSO SPEAK DYNAMIC: ECONORAM III™

If you want a dynamic memory, might as well get one that works right. Econoram III is inexpensive, completely assembled and tested, and ready to plug into your S-100 machine. Low power, 0 wait states with 8080 CPU, configured as two 4K blocks, fully socketed.

$149.00, assembled and tested only.

EDGE CONNECTORS

There are edge connectors, and there are **Edge Connectors.** These are the kind where the pins don't fall out, thanks to the bifurcated contacts. (We use the same connectors with our motherboards.)

#CK-1001: 100 pin edge connector with gold plated 3 level wrap posts. Mates with Altair/IMSAI peripherals. **$5 each or 5/$22.**

#CK-1002: Same as above, but with soldertail pins on 0.25" centers. (Mates with IMSAI motherboard). **$5 each or 5/$22.**

#CK-1003: Same as above, but with soldertail pins on 0.14" centers. (Mates with Altair motherboard). **$6 each or 5/$27.50.**

10 SLOT MOTHERBOARD

Whether implemented as an add-on to existing systems that need more room, or as the nucleus of a stand-alone system, this S-100 compatible motherboard fits the needs of the budget-minded enthusiast. Our price includes **all** edge connectors, along with active termination circuitry that promotes accurate and reliable data transfer. Lots of bypass caps and extra heavy power line traces contribute to efficient operation. Heavy duty epoxy glass board, with a solder mask for easy soldering.

#CK-015, $90.00. Kit form only.

18 SLOT MOTHERBOARD

All the same features and advantages of the 10 slot version, including our active termination circuitry. Complete with 18 edge connectors.

#CK-016, $124.00. Kit form only.

DEALER NOTE

We'd like to thank the ever-growing number of dealers who are spreading the Econoram word to their customers . . . you will be happy to know that we have **doubled** the capacity of our Compukit™ division in order to continue handling the massive response. We're glad you like what we're doing . . . and we're going to keep on doing it!

TERMS: Please allow up to 5% for shipping; excess refunded. Californians add tax. COD orders accepted with street address for UPS. For VISA*/Mastercharge* orders call our 24 hour order desk at **(415) 562-0636.** Prices good through cover month of magazine.

FREE FLYER: These are just a few of the items we carry for the computer enthusiast. We also stock a broad line of semiconductors, passive components, and hobbyist items. We will gladly send you a flyer describing our products upon receipt of your name and address.

GODBOUT

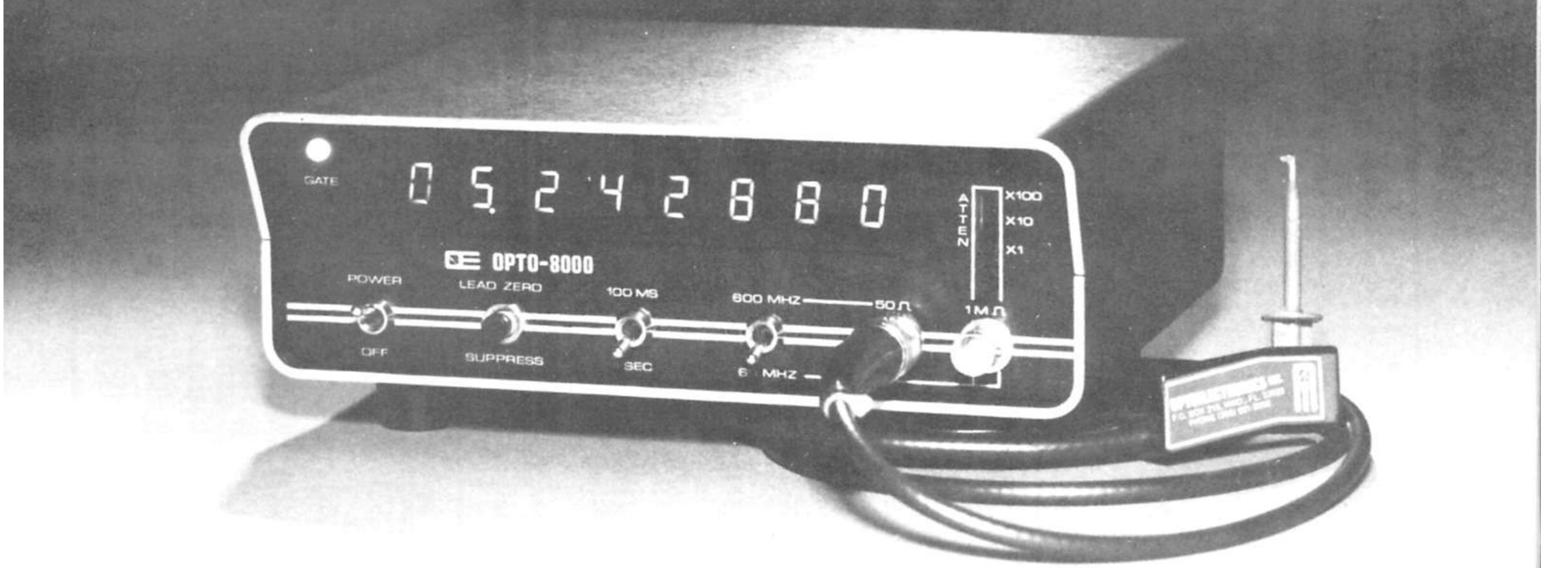
BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

G4

ask for us locally

600 MHZ. FREQUENCY COUNTER
±0.1 PPM TCXO

OPTO-8000.1



This new instrument has taken a giant step in front of the multitude of counters now available. The Opto-8000.1 boasts a combination of features and specifications not found in units costing several times its price. Accuracy of ±0.1 PPM or better — *Guaranteed* — with a factory-adjusted, sealed TCXO (Temperature Compensated Xtal Oscillator). **Even kits require no adjustment for guaranteed accuracy!** Built-in, selectable-step attenuator, rugged and attractive, black anodized aluminum case (.090" thick aluminum) with tilt bail. 50 Ohm and 1 Megohm inputs, both with amplifier circuits for super sensitivity and both diode/overload protected. Front panel includes "Lead Zero Blanking Control" and a gate period indicator LED. AC and DC power cords with plugs included.

OPTOELECTRONICS, INC.
5821 NE 14 Avenue
Ft. Lauderdale, FL 33334
Phones: (305) 771-2050   771-2051
Phone orders accepted 6 days, until 7 p.m.

03

master charge
BANKAMERICARD

SPECIFICATIONS:
Time Base—TCXO ±0.1 PPM GUARANTEED!
Frequency Range—10 Hz to 600 MHz
Resolution—1 Hz to 60 MHz; 10 Hz to 600 MHz
Decimal Point—Automatic
All IC's socketed (kits and factory-wired)
Display—8 digit LED
Gate Times—1 second and 1/10 second
Selectable Input Attenuation—X1, X10, X100
Input Connectors Type —BNC
Approximate Size—3"h x 7½"w x 6½"d
Approximate Weight—2½ pounds
Cabinet—black anodized aluminum (.090" thickness)
Input Power—9-15 VDC, 115 VAC 50/60 Hz
or internal batteries

OPTO-8000.1 Factory Wired	**$299.95**
OPTO-8000.1K Kit	**$249.95**

ACCESSORIES:

Battery-Pack Option—Internal Ni-Cad Batteries and charging unit	**$19.95**
Probes: P-100—DC Probe, may also be used with scope	**$13.95**
P-101—LO-Pass Probe, very useful at audio frequencies	**$16.95**
P-102—High Impedence Probe, ideal general purpose usage	**$16.95**
VHF RF Pick-Up Antenna-Rubber Duck w/BNC #Duck-4H	**$12.50**
Right Angle BNC adapter #RA-BNC	**$ 2.95**

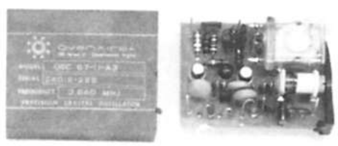
FC-50 — Opto-8000 Conversion Kits:
Owners of FC-50 counters with #PSL-650 Prescaler can use this kit to convert their units to the Opto-8000 style case, including most of the features.

FC-50 — Opto-8000	**Kit $59.95**
*FC-50 — Opto-8000F	**Factory Update $99.95**
FC-50 — Opto-8000.1 (w/TCXO)	**Kit $109.95**
*FC-50 — Opto-8000.1F	**Factory Update $149.95**

*Units returned for factory update must be completely assembled and operational

TERMS: Orders to U.S. and Canada, add 5% to maximum of $10.00 per order for shipping, handling and insurance. To all other countries, add 10% of total order. Florida residents add 4% state tax. C.O.D. fee: $1.00. Personal checks must clear before merchandise is shipped.

OVENAIRE ULTRA PRECISION CRYSTAL OSCILLATOR



Your computer is only as good as its clock. We have been fortunate in acquiring a lot of OVENAIRE precision crystal oscillators, Model OSC 67-11-A-3. The output frequency of these oscillators is 3.840 Mhz. This frequency readily divides into many useable frequencies with the use of standard SN7400 series ICs. Among the many frequencies are 640 Khz, 60 Khz, 32 Khz, 20 Khz, 10 Khz, 6 Khz, 1 Khz, 600 Hz, 100 Hz, 60 Hz, 50 Hz. and many more. We provide data showing the ICs needed to get these frequencies. The oscillator is precise to 2 parts per million, and is adjustable to even greater precision. Ideal for computers, frequency standars, clocks etc. This oscillator is a current production item, and the one piece price at the factory is $134.50. In lots of 100 the price is $49.80, so our price of $14.95 each is a fantastic bargain. 1 5/8"x 2"x 5/8". PC mount. Voltages required are 5 VDC and 12 VDC. Output is TTL compatible 5 VDC. Sketch at left shows the complete unit, and an inside view.

STOCK NO.5592K Ovenaire Precision Crystal Oscillator $14.95 ea. 2/28.00

NEW POWER TRANSFORMERS

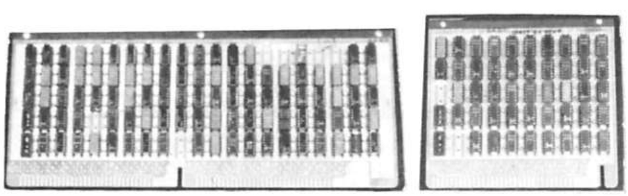
Tapped 115 V primary. Secondary either 12.5 or 14volts @ 2 A. 3 1/8"x3"x2½". 3 lbs.
STOCK NO.9031K $3.95 ea. 2/6.00

Primary 115V. Sec.1. 16.5V@ 1.5A. Sec.2 16V@ 3.5A. Sec.3 9.5V@ 3.5A. Sec.4, 130 V@2A. 10Lbs.
STOCK NO.6677K $10.95 ea. 2/20.00

SQUIRREL CAGE BLOWERS

Keep your valuable equipment from overheating. We have squirrel cage blowers, made by REDMOND, 115V, 60 Hz. .78 A. 3000 RPM. Removed from equipment.
STOCK NO.9325K 4 $9.95 ea. 2/18.00

WIRE WRAP BOARDS LOADED WITH 7400 SERIES ICs



Since last summer, we have been selling 2 wire wrap boards, Our Stock No. 6558K with approximately 100 sockets, and our Stock No. 6559K with approximately 45 sockets. These have been successful, based on your orders and reorders. We now have the same boards, but with the sockets still containing the original SN7400 series ICs that were used in the computer that these boards were designed for. We checked the value of these ICs, against the lowest price ICs in several Electronics magazine, and found that at the lowest possible surplus prices, the values of the ICs on the 100 socket board ran to over $40.00. A sample of some of the chips on the board we looked at are as f ollows: 74H87, 7486, 74107, 7451, 7400, 7404, 7495, 7493, 7492, 74193, 7489 and many others, to numerous to mention. Also on some boards, are a few linears, and phase locked loops. Not everyone needs every chip, but if you are working at all with TTL, this is a great opportunity to get an inventory of the most useful chips at a ridiculous price. We are selling the 100 socket board with about 100 chips, for $10.00 more than the board itself, and the 45 socket chip for $5.00 more than the board itself. We will also include with each board, 2 edge connectors with the 100 socket board, and 1 edge connector with the 45 socket board.

STOCK NO.6558K	75 to 100 socket board	$18.75 ea.	2/35.00
STOCK NO.6559K	45 to 50 socket board	$11.75	2/22.00
STOCK NO.6749K	75 to 100 socket wire wrap board with ICs and edge connectors	$28.75 ea.	2/55.00
STOCK NO.6750K	45 to 50 socket wire wrap board with ICs and edge connector	$16.75 ea.	2/32.00
STOCK NO.6603K	Edge connector for either board	$ 2.00 ea	3/5.00

VIDEOCUBE THE COMPUTER / TV INTERFACE

VIDEOCUBE is a self contained oscillator and modulator, which allows easy interface with any device having a video output, and a standard TV set. When properly used, the output of your video camera, video game or video output of your computer is displayed on channel 3. Easy switching from TV to monitor. The VIDEOCUBE was completely described in August issue of RADIO-ELECTRONICS. We supply a reprint of this article. Has FCC approval for radiation.
STOCK NO. 5500K Complete kit of parts with data. $13.95 2/26.00

MAIL ORDER

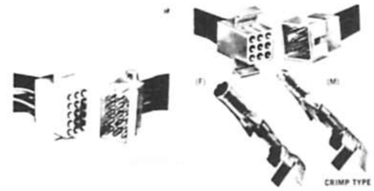
DELTA ELECTRONICS
P.O. BOX 2, 7 Oakland St.
Amesbury, Mass. 01913

Over the counter salesrooms

ATLANTA GA
DELTA ELECTRONIC HOBBIES
5151 Buford Hwy.
Doraville, Atlanta, Ga.

BOSTON MASS.
DELTA ELECTRONICS
WAREHOUSE OUTLET
590 Commonwealth Ave.
Boston, Mass.

D13



NYLON CONNECTORS

Mfg by Molex

Complete Connectors Per Pkg.	Type No.	Class	Description	Price Ea. Pkg.
5	1625-1PRT	Min.(.062")	1 Circuit	$1.75
3	1625-2PRT	"	2 Circuit	1.90
3	1625-3PRT	"	3 Circuit	2.10
2	1625-4PRT	"	4 Circuit	2.10
2	1625-5PRT	"	5 Circuit	2.20
2	1625-6PRT	"	6 Circuit	2.35
1	1649-8PRT	"	8 Circuit	1.55
1	1625-9PRT	"	9 Circuit	1.75
1	1625-12PRT	"	12 Circuit	1.90
1	1625-15PRT	"	15 Circuit	2.30
1	1625-24PRT	"	24 Circuit	3.25
1	1772-36PRT	"	36 Circuit	4.55
5	1619PRT	Std.(.093")	1 Circuit	1.75
3	1545PRT	"	2 Circuit	1.90
3	1396PRT	"	3 Circuit	2.10
2	1490PRT	"	4 Circuit	2.10
2	1653PRT	"	5 Circuit	2.20
2	1261PRT	"	6 Circuit	2.35
1	1292PRT	"	9 Circuit	1.80
1	1360PRT	"	12 Circuit	1.90
1	1375PRT	"	15 Circuit	2.45

Prototype hand tools combine efficiency with economy. Ideal for prototype or limited production runs.

HT 1919 for .093" pin dia. terminals ... $8.95 each

HT 1921 for .062" pin dia. terminals ... $8.95 each

Econo-Extractor removes terminal from nylon connector housing with smoothness and ease.

HT 2054 for extracting .093" pin dia. terminals ... $2.25 each

HT 2023 for extracting .062" pin dia. terminals ... $2.25 each

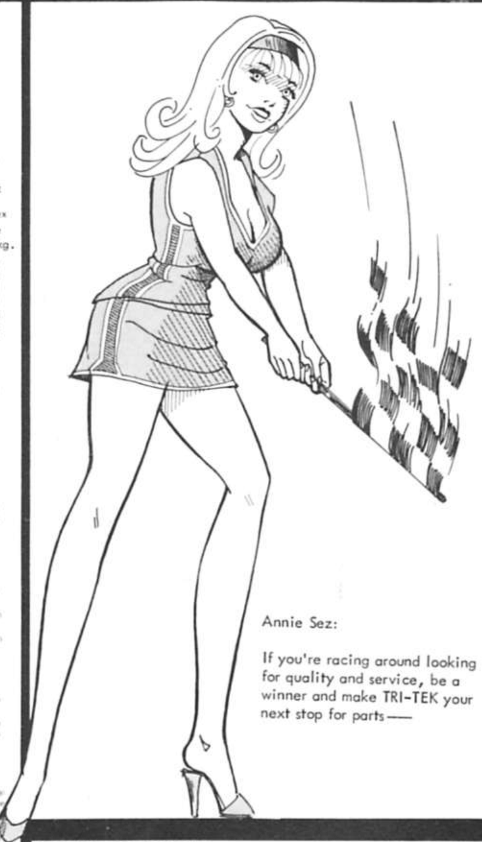
Deluxe ejector tools, spring loaded for simple, efficient removal of terminal from nylon connector housing extracts either male or female terminals of same pin diameter.

HT 2038 for extracting .093" pin dia. terminals ... $6.70 each

HT 1010-282 Replacement tip for HT 2038 ... $2.50 each

HT 2285 for extracting .062" pin dia. terminals ... $6.70 each

HT 1672-3 replacement tip for HT 2285 ... $2.50 each



Annie Sez:

If you're racing around looking for quality and service, be a winner and make TRI-TEK your next stop for parts—

P. C. BOARD TERMINAL STRIP

Molded body encloses postive screw activated clamp which will accomodate wire sizes 14-30 AWG. Contacts and pins are solder plated copper. Pins are on .200 inch (5.08mM) for standard P. C. mounting. 10Amp rating. Compare our prices before you buy.

4 pole	TS-2504	.99
8 pole	TS-2508	1.49
12 pole	TS-2512	2.19

S-100 BUS CONNECTORS (IMSAI TYPE)

Gold, Solder tail for Mother boards $4.50, 4/$17.00

Tin-Nickel, (NASGLO) Solder tail $3.75, 4/$14.00

Gold, wire-wrap................ $4.50, 4/$17.00

Tin-Nickel, (NASGLO) wire-wrap $3.75, 4/$14.00

HEX DARLINGTON ARRAY

MC1413P is a 16 pin DIP package with (6) 50V 500mA Darlington pairs.

MC1413P................................... $1.59

Specs/Apps.................................. .30¢

I.C. SOCKETS

Lo Profile Tin Solder Tail Dip Sockets

8 pin	10/$1.50	100/$14.00	1000/$120.00
14 pin	10/$1.70	100/$16.00	1000/$140.00
16 pin	10/$1.90	100/$18.00	1000/$160.00

VOLTAGE REGULATORS

7805-06-08-12-15-24 TO220	95¢ 5/$4.50
7905-06-08-12-15-24 TO 220	95¢ 5/$4.50
78L05A-12-15 4% 100 mA TO-92 Plastic	50¢
78H05KC 5V 5A TO-3	9.15
78H12KC 12V 5A TO-3	9.15
78H15KC 15V 5A TO-3	9.15
Lm317K 1.5A *Adjustable* TO-3	4.99
Lm317T 1.5A *Adjustable* TO-220	3.99
Lm317MP .5A *Adjustable* TO-202	13.95
TL430C Adjustable Zener-*Think About It*	1.50
TL497C Switching Reg. & Inductor	9.50
RCA CA 3085 100 mA *Adjustable*	.60

ADJUSTABLE NEGATIVE REGULATOR

LM337 is the compliment to the popular LM317 positive adjustable regulator. Capable of 1.5Amp from - 1.2V to - 37V.

LM337K.....(TO-3 Metal)................. $5.99

LM337T.....(TO-220 Plastic)............... $4.65

Specs and applications......................... 60¢

INTEGRATED CIRCUITS

555 Timer 8 pin mini-DIP	.49
741 Compensated OP-Amp 8 pin DIP	.37
LM 1889N RF Video Modulator	7.45
CA3130 Bipolar/Mos-FET Op Amp	1.19
CA3140 MOS-FET Op Amp, Bi-polar out	.99
LM3909 Lo Voltage Led Pulser	1.50
LM3911 Temp Control CHIP	1.50
Signetics 2504TA 1024 bit S.R. memory (1404A)..	$1.75
MCM 6571P Character Generator	11.95
MCM6571AP Character Generator	11.95
LM399H Temp Stabilized Zener	5.95
AF100-1CJ Active Filter, State Variable	7.50
LM2907N Tachometer F/V Converter	2.65
LM1812N Ultra-sonic Transceiver	9.15
LM1815 Adaptive Sense Amp for Tachometer	5.73
S2518B Hex 32 bit Shift Register...............	$2.95
TL170 TO-92 Hall effect switch w/spec sheets	1.25
MC14409P Telephone Rotary Pulser	10.98
MC14419P Touch Pad Converter for 14409	4.25
MC14411P Baud Rate Generator	11.98
MC14412VP CMOS Modem Chip	16.95
MM57109N Number Cruncher Micro	18.95
74C915 7 Segment to BCD Converter	2.99
74C922 16 Key Keyboard Encoder	6.35
74C923 20 key Keyboard Encoder	6.45
74C925 4 Decade Counter w/latches	12.00
74C926 4 Decade Counter w/carry	12.00
74C935-1 3½ Digit DVM CMOS Chip	16.98
9601 Retriggerable One shot..................	.50
MC4015P Hi Speed quad "D" low power TTL.....	$1.00

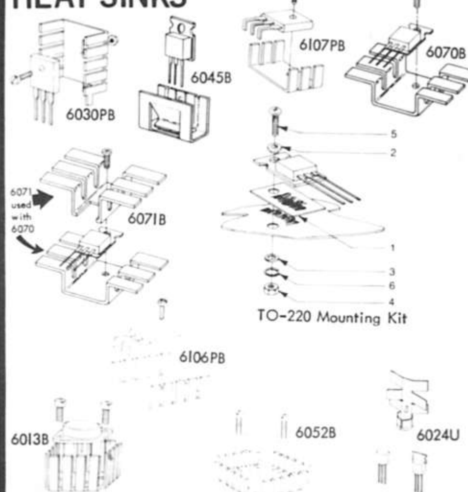
DATA AQUISITIONS SUBSYSTEM

ADC0817 is a 40 pin CMOS I.C. with an 8-bit analog-to-digital converter, a 16 channel multiplexer and micro processor compatible control logic. Converter features high impedance chopper stabilized comparator, voltage divider with analog switch tree and a sucessive approximation register. Latched Tri-State outputs for easy u-processor interface. Require only 15 mW of power from single 5V supply. Fast, 100uS conversion time.

ADC0817......(40 pin DIP)................. $29.88

Spec sheets................................. $1.00

HEAT SINKS



Economical 1 piece heat sinks for plastic power parts in TO-220 and Motorola cases 77, 90, 199, and TO-126. All are black anodized aluminum. "B" series is anodized after forming. "PB" series is anodized prior to forming.

THM 6030PB	Vertical	25¢, 5/$1.00, 10/$1.90
THM 6045B	Slip Over	30¢, 4/$1.00, 10/$2.00
THM 6070B	Extra Disipation Horiz.	35¢, 3/$1.00, 10/$3.00
THM 6071B	Top Hat for 6070	35¢, 3/$1.00, 10/$3.00
THM 6106PB	Flat With Fingers	30¢, 4/$1.00, 10/$2.00
THM 6107PB	Smaller Size Flat	25¢, 5/$1.00, 10/$1.90
THM 6013B	TO-3 Diamond	69¢, 4/$2.50, 10/$5.00
THM 6024-U	Unfinished TO-92	10/$1.00, 100/$5.00
THM-6052B	TO-3 Square	60¢, 5/$2.50, 10/$4.25

TO-220 6 Piece Mounting Kit - Handy Package...25¢

DIODES AND BRIDGES

IN4003 200 V 1 amp	12/$1.00
IN4004 400 V 1 amp	10/$1.00
IN4148 Hi Speed Signal	15/$1.00 100/$5.00
D-600 115 V, 100 mA Hi Speed Signal	20/$1.00
D2131 200 V, 25A Stud	85¢
D2135 400 V, 25A Stud	1.00
D2138 600 V, 25A Stud	1.55
D3289R 200 V, 160A Stud Anode	5.85
D3909-4 50 V, 45A Fast Recovery	2.00
IN4732A-47A 1W 5% Zeners	4/$1.00
13 Assorted *Brand New* Zener Diodes	1.00
50V 3 amp Epoxy Bridge	79¢
200V 30 amp Bridge	2.00
600V 4 amp Epoxy Bridge	1.49
600V 3 amp Stud Bridge	.89
SI-2 200V, 1.5A Gold Leads	15/$1.00
D1A-0030 30V DIAC	10/$1.00
100V 2.5 WATT ZENER	
IN5051A Epoxy Zener with heavy silver leads.....	25¢

MISCELLANEOUS

RG-174 Miniature 50 Ω coax	50'/4.25
WSU-30 Wire Wrap/unwrap tool	5.95
WSU-30M Modified Wrap/unwrap tool	6.95
BW-630 Battery Operated Wrap Tool	34.95
—Free Wire with any Wrap Tool—	
Miniature Square .05/100V Monolithic Cap	10/2.00
FND5002 .125" C.C. 7 Seg. Read-out..	49¢, 10/$3.95
2N4036 90V, 1A PNP Silicon TO-5	.50
2N6101 80V, 10A NON HI GAIN TO-220	.50
6.3 VCT, 1.2A Transformer F41X	2.49
12V, 1A Transformer with 6' Power Cord	2.88

Fairchild Linear Data Book. Huge volume has 17 chapters of data, applications and definitions. Probably the biggest volume of its type today........ $4.50ppd

Fairchild Bipolar Memory Data Book. ROMS, PROMS and RAMS are covered in this work with full engineering data. Price includes shipping!.......... $3.75

NEW NATIONAL DATA BOOK

1978 Edition of Nationals discrete devices data book. 9 big sections of data plus extensive glossary of terms and test diagrams. Covers bi-polar small signal and power devices as well as F.E.T.s............ $3.95

Raytheon Micro Computer Components Book. Covers Raytheons' offerings in the field from the micro-puter to memories and adapters................ $2.95ppd

TRI-TEK, INC.
7808 North 27th Avenue
Phoenix, Arizona 85021
(602) 995-9352

- Please give street address for UPS shipping when possible.
- C.O.D. NO parcel post C.O.D. UPS C.O.D. Add 85¢ to order.
- Any correspondence not connected with your order, please use separate sheet and include SASE for reply.
- Orders less than $10 ($15 foreign) please add $1 handling.
- Prices are subject to change without notice.
- Any refunds will be by check, not credit vouchers.
- Terms. Check, money order, credit card. Net 30 days to rated firms, schools and government agencies.
- If we should be temporarily out of stock on an item, it will be placed on back order. If we cannot ship in 30 days, you will be notified of the expected shipping date and furnished with a postage paid card with which to cancel your order if desired.
- We pay surface shipping only in USA, Canada and Mexico.
- For premium shipping (first class, special handling, etc.) add extra. Excell will be refunded.
- Foreign orders (except Canada and Mexico) estimate and add shipping. Excess will be refunded.

T1

Charge card telephone orders ($20 min.) will be accepted 9-5:30 P.M. except weekends. Telephone 995-9352. No collect calls please.

DIODES/ZENERS

1N914	100v	10mA	.05
1N4005	600v	1A	.08
1N4007	1000v	1A	.15
1N4148	75v	10mA	.05
1N753A	6.2v	z	.25
1N758A	10v	z	.25
1N759A	12v	z	.25
1N4733	5.1v	z	.25
1N5243	13v	z	.25
1N5244B	14v	z	.25
1N5245B	15v	z	.25

SOCKETS/BRIDGES

8-pin	pcb	.25	ww	.45
14-pin	pcb	.25	ww	.40
16-pin	pcb	.25	ww	.40
18-pin	pcb	.25	ww	.75
22-pin	pcb	.45	ww	1.25
24-pin	pcb	.35	ww	1.10
28-pin	pcb	.35	ww	1.45
40-pin	pcb	.50	ww	1.25

Molex pins .01 To-3 Sockets .45
2 Amp Bridge 100-prv 1.20
25 Amp Bridge 200-prv 1.95

TRANSISTORS, LEDS, etc.

2N2222A	NPN (2N2222 Plastic .10)	.15
2N2907A	PNP	.15
2N3906	PNP (Plastic)	.10
2N3904	NPN (Plastic)	.10
2N3054	NPN	.35
2N3055	NPN 15A 60v	.50
T1P125	PNP Darlington	.35
LED Green, Red, Clear, Yellow		.15
D.L.747	7 seg 5/8" High com-anode	1.95
XAN72	7 seg com-anode (Red)	1.25
MAN71	7 seg com-anode (Red)	1.25
MAN3610	7 seg com-anode (Orange)	1.25
MAN82A	7 seg com-anode (Yellow)	1.25
MAN74A	7 seg com-cathode (Red)	1.50
FND359	7 seg com-cathode (Red)	1.25

C MOS

4000	.15
4001	.15
4002	.20
4004	3.95
4006	.95
4007	.35
4008	.95
4009	.45
4010	.45
4011	.20
4012	.20
4013	.40
4014	.95
4015	.90
4016	.35
4017	1.10
4018	1.10
4019	.50
4020	.85
4021	1.00
4022	.85
4023	.25
4024	.75
4025	.30
4026	1.95
4027	.50
4028	.95
4030	.35
4033	1.50
4034	2.45
4035	1.25
4040	1.35
4041	.69
4042	.95
4043	.95
4044	.95
4046	1.75
4049	.45
4050	.45
4066	.95
4069	.40
4071	.35
4081	.70
4082	.45
MC14409	14.50
MC14419	4.85

— T T L —

7400	.15
7401	.15
7402	.20
7403	.20
7404	.15
7405	.25
7406	.35
7407	.55
7408	.25
7409	.15
7410	.10
7411	.25
7412	.30
7413	.35
7414	1.10
7416	.25
7417	.40
7420	.15
7426	.30
7427	.45
7430	.15
7432	.30
7437	.30
7438	.35
7440	.25
7441	1.15
7442	.45
7443	.65
7444	.45
7445	.65
7446	.95
7447	.95
7448	.65
7450	.25
7451	.25
7453	.20
7454	.25
7460	.40
7470	.45
7472	.40

7473	.25
7474	.30
7475	.35
7476	.40
7480	.55
7481	.75
7483	.95
7485	.75
7486	.25
7489	1.35
7490	.55
7491	.95
7492	.95
7493	.35
7494	.75
7495	.60
7496	.80
74100	1.15
74107	.35
74121	.35
74122	.55
74123	.55
74125	.45
74126	.35
74132	1.35
74141	.90
74150	.85
74151	.65
74153	.75
74154	.95
74156	.95
74157	.65
74161	.85
74163	.85
74164	.60
74165	1.50
74166	1.35
74175	.80

74176	1.25
74180	.75
74181	2.25
74182	.95
74190	1.75
74191	1.05
74192	.75
74193	.85
74194	1.25
74195	.95
74196	1.25
74197	1.25
74198	2.35
74221	1.00
74367	.85
75108A	.35
75110	.35
75491	.50
75492	.50
74H00	.15
74H01	.25
74H04	.20
74H05	.20
74H08	.35
74H10	.35
74H11	.35
74H15	.45
74H20	.30
74H21	.25
74H22	.40
74H30	.20
74H40	.25
74H50	.25
74H51	.25
74H52	.15
74H53J	.25
74H55	.20

74H72	.45
74H101	.75
74H103	.75
74H106	.95
74L00	.25
74L02	.25
74L03	.30
74L04	.30
74L10	.30
74L20	.35
74L30	.45
74L47	1.95
74L51	.45
74L55	.65
74L72	.45
74L73	.40
74L74	.45
74L75	.55
74L93	.55
74L123	.85
74S00	.35
74S02	.35
74S03	.30
74S04	.30
74S05	.35
74S08	.35
74S10	.35
74S11	.35
74S20	.35
74S40	.20
74S50	.20
74S51	.25
74S64	.20
74S74	.35
74S112	.60
74S114	.65

74S133	.40
74S140	.55
74S151	.30
74S153	.35
74S157	.75
74S158	.30
74S194	1.05
74S257 (8123)	1.05
74LS00	.25
74LS01	.35
74LS02	.35
74LS04	.30
74LS05	.45
74LS08	.25
74LS09	.35
74LS10	.35
74LS11	.35
74LS20	.25
74LS21	.25
74LS22	.25
74LS32	.40
74LS37	.35
74LS40	.45
74LS42	1.10
74LS51	.50
74LS74	.65
74LS86	.65
74LS90	.95
74LS93	.95
74LS107	.85
74LS123	1.00
74LS151	.95
74LS153	1.20
74LS157	.85
74LS164	1.90
74LS367	.75
74LS368	.75
74C04	.25
74C151	2.25

9000 SERIES

9301	.85	95H03	1.10
9309	.35	9601	.45
9322	.75	9602	.45

MICRO'S, RAMS, CPU'S, ETC.

74S188	3.00
1702A	4.50
MM5314	3.00
MM5316	3.50
2102-1	1.45
2102L-1	1.75
TR1602B	4.50
TMS 4044-45NL	14.50
8080AD	12.00
8T13	1.50
8T23	1.50
8T24	2.00
8T97	1.00
2107B-4, A	4.00
2708	11.50

LINEARS, REGULATORS, etc.

MCT2	.95
8038	3.95
LM201	.75
LM301	.45
LM308 (Mini)	.95
LM309H	.65
LM309K (340K-5)	.85
LM310	1.15
LM311D (Mini)	.75
LM318 (Mini)	.95
LM320K5(7905)	1.65
LM320K12	1.65

LM320T5	1.65
LM320T12	1.65
LM320T15	1.65
LM324N	.95
LM339	.95
7805 (340T5)	.95
LM340T12	1.00
LM340T15	1.00
LM340T18	1.00
LM340T24	.95
LM340K12	1.65

LM340K15	1.25
LM340K18	1.25
LM340K24	.95
78L05	.75
78L12	.75
78L15	.75
78M05	.75
LM373	2.95
LM380 (8-14 PIN)	.95
LM709 (8,14 PIN)	.25
LM711	.45

LM723	.50
LM725N	2.50
LM739	1.50
LM741 (8-14)	.25
LM747	1.10
LM1307	1.25
LM1458	.95
LM3900	.50
LM75451	.65
NE555	.50
NE556	.95
NE565	.95
NE566	1.75
NE567	1.35

INTEGRATED CIRCUITS UNLIMITED 19

7889 Clairemont Mesa Boulevard, San Diego, California 92111
(714) 278-4394 *(Calif. Res.)*

All orders shipped prepaid — No minimum
Open accounts invited — COD orders accepted

Discounts available at OEM Quantities — California Residents add 6% Sales Tax
All IC's Prime/Guaranteed. All orders shipped same day received.

24 Hour Toll Free Phone 1-800-854-2211 — **American Express / BankAmericard / Visa / MasterCharge**

SPECIAL DISCOUNTS

Total Order	Deduct
\$35 - \$99	5%
\$100 - \$300	10%
\$301 - \$1000	15%
\$1000 - Up	20%

THE CHIPS ARE DOWN!

8K NOW JUST $149 ASSEMBLED

Thinker Toys™ brings down the high cost of adding big memory capacity to your S-100 system!

The ECONORAM III* 8Kx8 (by Morrow's Micro-Stuff) comes fully assembled, burned in, tested and fully warranted for one full year—*for just $149!*

It's configured as two individually addressable 4K blocks. And it typically consumes less than one-half the power of any competitively-priced memory.

Obviously, our new ECONORAM III* 8Kx8 isn't just another cheap kit. It's a design breakthrough in dynamic memory that gives you guaranteed reliability with tremendous savings in cost and power.

You save on cost and power because ECONORAM III* is the first fully reliable dynamic memory. It's unique SynchroFresh™ refresh circuitry weaves itself into the natural timing of the S-100 bus to provide simple, invisible and absolutely reliable refresh . . . with fewer parts, less power, lower cost and no interference with computer speed or timing.

You just plug ECONORAM III* into your S-100 bus and run perfectly for at least one full year. Or your money fully refunded. It's that simple.

So don't let the high cost of memory keep your system down. ECONORAM III* 8Kx8 memory is available now at leading computer shops. Or ask your nearest computer shop to order it for you.

Or, if unavailable locally, write direct to Thinker Toys™ for specs and ordering. BAC/MC orders can be placed by phone to 415-527-7548 (10-4 PST). Add $3 for handling; Cal. res. add tax.

Morrow's Micro-Stuff for

Thinker Toys™

1201 10th Street Berkeley, CA 94710

***ECONORAM is a trademark of Godbout Electronics**

George Morrow-Inventor & Designer

Herb Waite looks up from behind his copy of KB.

Who's Behind the Kilobaud?

In between selling subscriptions and seeing the other exhibits at computer shows, Wayne snaps pictures of people you have been seeing and will be seeing at shows. How many of them can you recognize behind the Kilobauds?

Still waiting for the answer from last month's clues? Drop by the PerSci booth at the next computer show and talk with Herb . . . he'll tell you about one of the most popular disk drives available today as well as confirm that KILOBAUD is a favorite magazine of his . . . and thousands of other computerists.

Kilobaud is the most read magazine in the microcomputer field. It hasn't been difficult to find key industry people reading KB!

kilobaud

WHO'S BEHIND THE KILOBAUD?

Who's This?

In addition to being a frequent author in KILOBAUD, the chap behind the KILOBAUD is also the owner of a microcomputer store. As a further hint, the store is in a small city of about 40,000, even so, business is growing nicely and the store is thriving. Not bad when you consider there are two other computer stores in the same town . . . and a fourth in the works!

READ IT YOURSELF!

If you are not yet a subscriber to Kilobaud, we need you and you need us. There is a lot of data in Kilobaud that you will want to have on hand . . . the magazine is like a continuing encyclopedia of microcomputing and programs. You never know when you are desperately going to need something from a past issue . . . and you'll want it immediately. The cost per year is not significant . . . $15 at present for $24 worth of magazines.

You may have noticed that Kilobaud has more articles than any of the other magazines. If you'll keep track of how long it takes you to read Kilobaud as compared to the other computer magazines, you'll see you are getting a great bargain at $15 per year.

SUBSCRIBE TODAY!

For Instant Subscriptions Call Our Toll Free Number

1-(800) 258-5473

Have your credit card handy!

Kilobaud ● Peterborough NH 03458

Kb book nook Kb book nook

beginner's·introductory

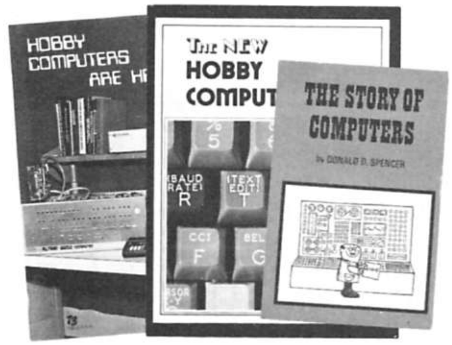
●**THE STORY OF COMPUTERS** by Donald D. Spencer is to computer books what *Dick and Jane* is to novels . . . extremely elementary, gives the non-computerist a fair idea of what the hobbyist is talking about when he speaks computer lingo. Attempts to explain what computers are and can do to a spouse, child or any un-electronics-minded friend. $4.95.*

●**HOBBY COMPUTERS ARE HERE** If you (or a friend) want to come up to speed on how computers work . . . hardware and software . . . this is an excellent book. It starts with the fundamentals and explains the circuits, the basics of programming, along with a couple of TVT construction projects, ASCII-Baudot, etc. This book has the highest recommendations as a teaching aid for newcomers. $4.95.*

●**THE NEW HOBBY COMPUTERS!** This book takes it from where "Hobby Computers Are Here" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards, a Baudot monitor/editor system, an audible logic probe for finding those tough problems, a ham's computer, a computer QSO machine . . . and much, much more! $4.95.*

●**HOME COMPUTERS: 2^{10} Questions & Answers** by Rich Didday. Two books aimed exclusively at the novice computer hobbyist/home computer user. Written in a rather unusual style which has a beginner asking questions which are answered by a person with a substantial background in computers and personal computing. The questions are just the kind beginners come up with . . . and the answers are presented in easy-to-understand terms (usually with a diagram to illustrate the point). Both the hardware and software aspects of home computing are covered from A to Z. An index in both books makes them ideal as reference material *for anyone.* Volume I: Hardware — $7.95*; Volume 2: Software — $6.95*.

●**MICROCOMPUTER PRIMER** by Mitchell Waite and Michael Pardee. Describes basic computer theory, explains numbering systems, and introduces the reader to the world of programming. Describes the world of microcomputing in "real world" terminology. No better way of getting involved with the exciting new hobby of microcomputing. $7.95.*



●**MICROPROCESSORS FROM CHIPS TO SYSTEMS** by Rodnay Zaks is a complete and detailed introduction to microprocessors and microcomputer systems. No preliminary knowledge of computers or microprocessors is required to read this book, although a basic engineering knowledge is naturally an advantage. Intended for all wishing to understand the concepts, techniques and components of microprocessors in a short time. $9.95.*

●**INTRODUCTION TO MICROPROCESSORS** by Charles Rockwell of MICROLOG is an ideal reference for the individual desiring to understand the hardware aspects of microprocessor systems. Describes the hardware details of computer devices in terms the beginner can understand, instead of treating the micro chip as a "black box." General information about hardware systems is provided. Specific systems are not described and programming is only briefly discussed. $17.50 US and Canada, $20 elsewhere.*

●**AN INTRODUCTION TO MICROCOMPUTERS, VOLS. 1 AND 2** by Adam Osborne Associates, are references dealing with microcomputer architecture in general and specifically with details about most of the common chips. These books are not software-oriented, but are invaluable for the hobbyist who is into building his own interfaces and processors. Volume 1 is dedicated to general hardware theory related to micros, and Volume 2 discusses the practical details of each micro chip. (Detailed review in Kilobaud #2) Volume 1 — $7.50*; Volume II — $12.50.*

● **HOME COMPUTERS: A BEGNINNERS GLOSSARY AND GUIDE** this book is intended as a quick reference source for beginners. Included is a general introduction to microcomputers, a simple application & sample system, the history of microcomputers & their uses, and an introduction to same actual equipment. A chapter on number systems includes a number conversion chart, binary arithmetic from conversions to divisions, and a discussion of octal and hexadecimal numbers. A good background to read technical literature and computer equipment specifications. $6.95.*

computer games

●**WHAT TO DO AFTER YOU HIT RETURN** PCC's first book of computer games . . . 48 different computer games you can play in BASIC . . . programs, descriptions, muchly illustrated. Lunar landing, Hammurabi, King, Civel 2, Qubic 5, Taxman, Star Trek, Crash, Market, etc. $8.00.*

●**SCELBI'S GALAXY GAME FOR THE "6800"** Here's a new twist in computer games by Robert Findley/Raymond Edwards. "Galaxy" pits the operator of a spaceship against alien craft, as well as such variables as speed, time, and ammunition. No two games are the same! $14.95.*

● **101 BASIC COMPUTER GAMES** Okay, so once you get your computer up and running in BASIC, then what? Then you need some programs in BASIC, that's what. This book has 101 games for you, from very simple to real buggers. You get the games, a description of the games, the listing to put in your computer and a sample run to show you how they work. Fun. Any one game will be worth more than the price of the book for the fun you and your family will have with it. $7.50.*

●**SCELBI'S FIRST BOOK OF COMPUTER GAMES** Need a game for your 8008 or 8080 microprocessor? Has three popular games, "Space Capture," "Hexpawn," and "Hangman." Complete flowcharts, logic description, program listing, and instructions are provided. A must for the game freak! $14.95.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to:
Kilobaud Book Department Peterborough NH 03458
Be sure to include check or detailed credit card information

* *Add $1.00 shipping & handling charge for each order.* Note: Prices subject to change on books not published by 73 Magazine.

software·programming

●**DISCOVERING BASIC — A Problem Solving Approach** by Robert E. Smith deals with progressively more complex problems which allow the reader to discover the vocabulary of BASIC language as he develops skill and confidence in putting it to work. Clear and concise explanations. Problems used cover a wide range of interests — insurance, geometry, puzzles, economics, etc. $6.85.*

● **BASIC New 2nd Edition.** by Bob Albrecht. Self-teaching guide to the computer language you will need to know for use with your microcomputer. This is one of the easiest ways to learn computer programming. $4.95.*

●**A QUICK LOOK AT BASIC** by Donald D. Spencer. A perfect reference for the beginning programmer. Assumes that the reader has no previous programming experience and is a self-teaching guide for the individual desiring to learn the fundamentals of BASIC. $4.95.*

●**MY COMPUTER LIKES ME . . . WHEN I SPEAK BASIC** An introduction to BASIC . . . simple enough for your kids. If you want to teach BASIC to anyone quickly, this book is the way to go. $2.00.*

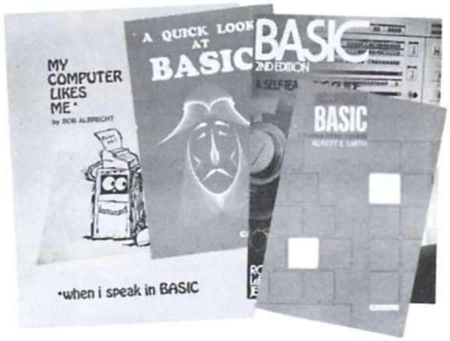
●**FUN WITH COMPUTERS AND BASIC** by Donald D. Spencer, contains an easy-to-understand explanation of the BASIC Programming Language and is intended for persons who have had no previous exposure to computer programming. Over half the book is devoted to problems using games, puzzles, and mathematical recreations. A superior book for self-teaching and learning computer programming. $6.95.*

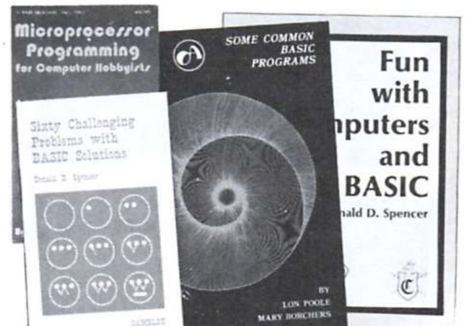
● **SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS** by Donald Spencer, provides the serious student of BASIC programming with interesting problems and solutions. No knowledge of math above algebra required. Incudes a number of game programs, as well as programs for financial interest, conversions and numeric manipulations. $6.95.*

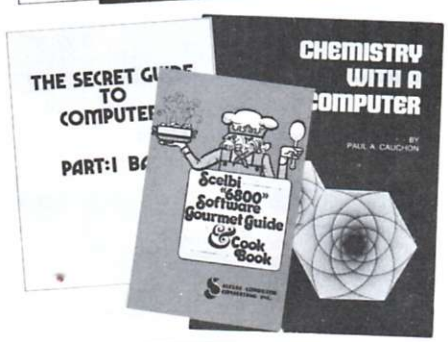
●**THE SECRET GUIDE TO COMPUTERS** Parts 1, 2, and 3 by Russ Walter. **Part One** describes computers in general, and after reading for ten minutes you will be writing simple BASIC programs! **Part Two** discusses computer applications. It's one thing to master the syntax of a language such as BASIC and another to solve problems using the new tool. **Part Three** describes programming languages. Ever heard of APL and QLISP? BASIC is not the only language used to program computers. 7th Edition. Part I - $2.75*; Part II — $2.50*; Part III — $3.50.*

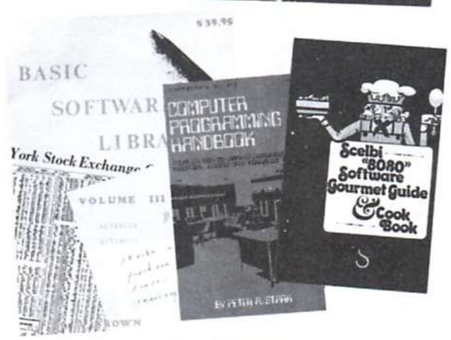
●**SOME COMMON BASIC PROGRAMS** published by Adam Osborne & Associates, Inc. Perfect for non-technical computerists requiring ready-to-use programs. Business programs, plus miscellaneous programs. Invaluable for the user who is not an experienced programmer. All will operate in the standalone mode. $7.50 paperback.*

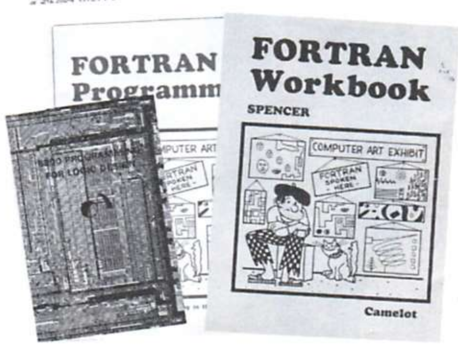
●**Scientific Research Instruments' BASIC SOFTWARE LIBRARY** is a complete do-it-yourself kit. Written in everybody's BASIC immediately executable in ANY computer with at least 4K, no other peripherals needed. Vol. I contains business and recreational programs and is 300 pages. Vol. II is 260 pages and contains math, engineering, statistics and plotting programs. Vol. III contains money managing, advanced business programs such as billing, A/R, inventory, payroll, etc. Vol. IV contains general purpose programs like loans, rates, retirement, plus games: Poker, Enterprise (take charge while Capt. Kirk is away), Football and more! Vol. V is filled with experimenter's programs including games, pictures and misc. problems like "logic." Vols. I & II $24.95,* Vol. III $39.95,* Vol. IV & V $9.95 each.*











●**MICROPROCESSOR PROGRAMMING for Computer Hobbyists** by Neill Graham is for the hobbyist interested in intermediate and advanced techniques of programming and data structuring. Written to take up where the computer manufacturers' instruction manuals and the introductory programming language texts leave off. $8.95.*

● **ADVANCED BASIC** — Applications and Problems by James Coan is for those who want to extend their expertise with BASIC. Offers advanced techniques and applications. $6.95.*

● **8080 PROGRAMMING FOR LOGIC DESIGN** Ideal reference for an in-depth understanding of the 8080 processor. Application-oriented and the 8080 is discussed in light of replacing conventional, hard-wired logic. Practical design considerations are provided for the implementation of an 8080-based control system. $7.50.*

●**8080 SOFTWARE GOURMET GUIDE AND COOKBOOK** If you have been spending too much time developing simple routines for your 8080, try this new book by Scelbi Computing and Robert Findley. Describes sorting, searching, and many other routines for the 8080 user. $9.95.*

● **6800 PROGRAMMING FOR LOGIC DESIGN** Oriented toward the industrial user, this book describes the process by which conventional logic can be replaced by a 6800 microprocessor. Provides practical information that allows an experimenter to design a complete micro control system for the "ground up." $7.50.*

●**6800 SOFTWARE GOURMET GUIDE & COOKBOOK** If you have been spending too much time developing routines for your 6800 microprocessor, try the new book by Scelbi Computing and Robert Findley. Describes sorting, searching, and many other routines for the 6800 user. $9.95.*

●**FORTRAN PROGRAMMING** by Donald Spencer. FORTRAN was designed for complex numeric calculations; and possesses extended I/O capability. It is easily learned, as it is an English-like computer language. $7.50.*

● **FORTRAN WORKBOOK** by Donald Spencer. Provides practical examples and problems to solve. Flowcharting is also discussed. Almost all micros support BASIC — it won't be long before FORTRAN is commonplace. $3.95.*

●**CHEMISTRY WITH A COMPUTER** by Paul A. Cauchon, contains a collection of tutorial, simulation and problem-generation computer programs. Usable with almost any chemistry course in the high school or college level. $9.95.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to:
Kilobaud Book Department   Peterborough NH 03458
Be sure to include check or detailed credit card information

* *Add $1.00 shipping & handling charge for each order.* Note: Prices subject to change on books not published by 73 Magazine.

kB book nook kB book nook

KB book nook KB book nook

hardware

● **MICROPROCESSOR INTERFACING TECHNIQUES** by Austin Lesea & Rodnay Zaks will teach you how to interconnect a complete system and interface it to all the usual peripherals. It covers hardware and software skills and techniques, including the use and design of model buses such as the IEEE 488 or $100. $9.95.*

● **TTL COOKBOOK** by Donald Lancaster. Explains what TTL is, how it works, and how to use it. Discusses practical applications, such as a digital counter and display system, events counter, electronic stopwatch, digital voltmeter, and a digital tachometer. $8.95.*

● **CMOS COOKBOOK** by Don Lancaster. Details the application of CMOS, the low power logic family suitable for most applications presently dominated by TTL. Required reading for every serious digital experimenter! $9.95.*

● **TVT COOKBOOK** by Don Lancaster, describes the use of a standard television receiver as a microprocessor CRT terminal. Explains and describes character generation, cursor control and interface information in typical, easy-to-understand Lancaster style. $9.95.*

● **BUILD-IT BOOK OF DIGITAL ELECTRONIC TIMEPIECES** by Robert Haviland is a data-packed guide to building every timekeeping device you can imagine: rugged shipboard clocks, second-splitting digital IC chronometers, decorator digital clocks, a precision timer, a frequency-period meter, a tide and moon clock, an automatic alarm setter, etc. Including full-size printed circuit board layouts. $6.95.*

● **THE "COMPULATOR" BOOK – Building Super Calculators & Minicomputer Hardware with Calculator Chips** by R. P. Haviland, provides ideas, design info and printed circuit boards for calculator chip projects, measure time, tie in with a Teletype to create a virtually infinite memory system, and countless other functions. $7.95.*

TEST EQUIPMENT LIBRARY

● **VOL I COMPONENT TESTERS** Build your own test equipment and save a bundle (and have a lot of fun). Volume I of the 73 Test Equipment Library shows you how to build and use transistor testers (8 of 'em), three diodes testers, 3 IC testers, 9 voltmeters and VTVMs, 8 ohmmeter, 3 inductance meters, and a raft of other gadgets for checking temperature, crystals, Q, etc. $4.95.*

● **VOL II AUDIO FREQUENCY TESTERS** If you're into audio such as digital cassette recording, RTTY, Baudot vs ASCII, SSTV, SSB, Touchtone or even hi-fi you'll want to have this book full of home built test equipment projects. Volume II $4.95.*

● **VOL III RADIO FREQUENCY TESTERS** This is of more interest to hams and CBers: test equipment you can build for checking out transmitters and receivers, signal generators, noise generators, crystal calibrators, GDOs, dummy loads. $4.95.*

● **VOL. IV IC TEST EQUIPMENT** Become a trouble-shooting wizard. All you need to know about pulse, audio and sync generators, frequency counters, digital component testers, logic probes and more! Plus a cumulative index for all four volumes of the 73 Test Equipment Library. $4.95.*

general

● **COMPUTER DICTIONARY** by Donald D. Spencer. Defines words and acronyms used by computerists in a clear, easy to understand style. This reference is a must for the individual getting started in the world of microcomputers. $5.95.*

● **BRAND NEW DICTIONARY** This microcomputer dictionary fills the need to become quickly acquainted with the terminology and nomenclature of the revolution in computers. There is also a comprehensive electronics/computer abbreviations and acronyms section. $15.95.*

● **THE UNDERGROUND BUYING GUIDE** Here is a handy guide for the electronics enthusiast. Over 600 sources of equipment and literature are provided. Cross-referenced for ease of use. Electronic publishing houses are also listed. $5.95 each.*

● **1976 PERIODICAL GUIDE FOR COMPUTERISTS** is a 20-page book which indexes over 1,000 personal computing articles for the entire year of 1976 from Byte, Creative Computing, Digital Design, Dr. Dobbs Journal, EDN, Electronic Design, Electronics, Interface Age, Microtrek, Peoples Computer Company, Popular Electronics, QST, Radio Electronics, SCCS Interface and 73 Amateur Radio. Price $3.00.* New January – June 1977 Edition (includes Kilobaud) – $3.00.*

● **TYCHON'S 8080 OCTAL CODE CARD** Slide rule-like aid for programming and debugging 8080 software contains all the mnemonics and corresponding octal codes. Also available, Tychon's *8080 Hex Code Card*, same as above only has hex codes instead of octal. $3.00 each.*

amateur radio books

● **NOVICE STUDY GUIDE** The most complete Novice study guide available. It is brand new. This is not only invaluable for anyone wanting to get started in amateur radio, but also it is about the only really simple book on the fundamentals of electricity and electronics. $4.95.*

● **GENERAL CLASS STUDY GUIDE** Takes over on theory where the Novice book leaves off. You'll need to know the electronic theory in this to work with computers and you'll not find an easier place to get the information. $5.95.*

● **SSTV HANDBOOK** This excellent book tells all about it, from its history and basics to the present state-of-the-art techniques. Hardbound $7,* Softbound $5.*

● **VHF ANTENNA HANDBOOK** This new handbook details the theory, design and construction of hundreds of different VHF and UHF antennas. Packed with fabulous antenna projects you can build. $4.95.*

● **WEATHER SATELLITE HANDBOOK** Simple equipment and methods for getting good pictures from the weather satellite. Dr. Taggart WB8DQT $4.95.*

● **THE NEW RTTY HANDBOOK** is a brand new 1977 edition and the only up-to-date RTTY book available. The state-of-the-art has been changing radically and has made all previous RTTY books obsolete. It has the latest circuits, great for the newcomer and the expert alike. $5.95.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to:
Kilobaud Book Department Peterborough NH 03458 Be sure to include check or detailed credit card information

Add $1.00 shipping & handling charge for each order. Note: Prices subject to change on books not published by 73 Magazine.

NEW ADDITIONS TO THE
KB BOOK NOOK

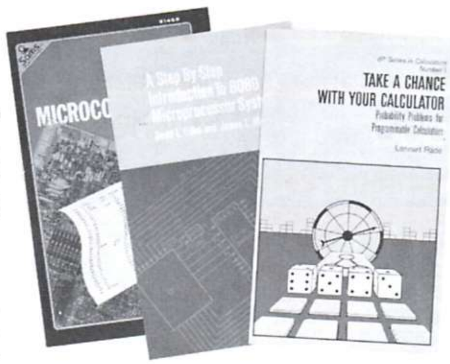
● **MICROPROCESSORS FROM CHIPS TO SYSTEMS** by Rodnay Zaks is a complete and detailed introduction to microprocessors and microcomputer systems. No preliminary knowledge of computers or microprocessors is required to read this book, although a basic engineering knowledge is naturally an advantage. Intended for all wishing to understand the concepts, techniques and components of microprocessors in a short time. $9.95.*

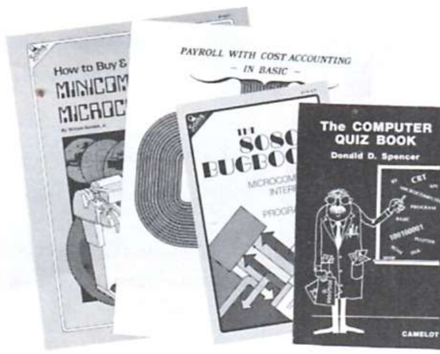
● **YOUR OWN COMPUTER** by M. Waite and M. Pardee. The personal computer has been touted as the next consumer product. But most individuals still wonder why. Much technical material has been written but there is little material for the average individual without an extensive background in electronics. This book removes the stigma of complexity that surrounds the computer and has succeeded in providing a simple easy-to-understand guide to these units. $1.95.*

● **BEGINNING BASIC** by Dr. Paul Chirlian provides a really basic BASIC that covers all the topics in simple, easy-to-understand language. Nothing is left out, everything is presented in clear, step-by-step fashion. This book will make a good BASIC programmer of any reader. $9.95.*

● **THE COMPUTER QUIZ BOOK** by Donald D. Spencer is written for readers who would like to test themselves on basic computer concepts. It may be used effectively by students, teachers, laymen, programmers, personal computer users, or anyone else interested in checking their knowledge of computer concepts. $5.95.*

● **PAYROLL WITH COST ACCOUNTING – IN BASIC** by L. Poole & M. Borchers includes program listings with remarks, descriptions, discussion of the principle behind each program, file layouts, and a complete user's manual with step-by-step instructions, flow charts, and simple reports and CRT displays. Payroll and cost accounting features include separate payrolls for up to 10 companies, time-tested interactive data entry, easy correction of data entry errors, job costing (labor distribution), check printing with full deduction and pay detail, and 16 different printed reports, including W-2 and 941. Price only $12.50.*





● **HOW TO BUY & USE MINICOMPUTERS AND MICROCOMPUTERS** by Wm. Barden, Jr. This book discusses these smaller brethren of computers and shows how the reader can become a part of the revolution – how he can own and use a functioning computer system in his home to do a variety of practical or recreational tasks. $9.95.*

● **THE 8080A BUGBOOK–MICROCOMPUTER INTERFACING AND PROGRAMMING** is written for the 8080 user who has a knowledge of digital elements and operations. This book will be invaluable, as it explains the fundamental tasks of microcomputer interfacing and the associated microcomputer I/O programming for 8080-based microcomputers. Only $9.95.*

● **HOW TO PROGRAM MICROCOMPUTERS** by Wm. Barden, Jr. Here is a guide to assembly language programming of the Intel 8080, Motorola MC6800, and MOS Technology MCS6502 microprocessors. It is written especially for beginning programmers with hobbyist microcomputers based on one of these three chips. The topics covered range from data manipulations at the bit level up to data handling of tables and lists, and from simple adds and subtracts up to floating-point operations. $8.95.*

● **TAKE A CHANCE WITH YOUR CALCULATOR** Lennart Rade wrote this book to help you to discover the world of probability with your programmable calculator. You will need NO previous experience either in probability theory or in programming to learn both from this book. $8.95.*

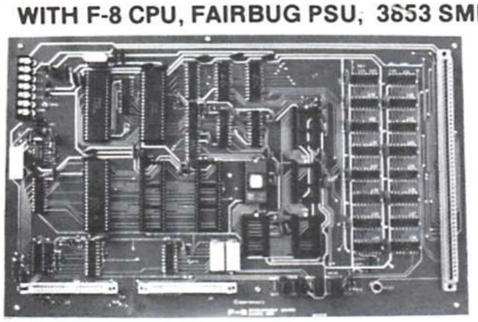
● **A STEP BY STEP INTRODUCTION TO 8080 MICROPROCESSOR SYSTEMS** by D.L. Cohn & J.L. Melsa. This microprocessor book is written for people who don't know anything about microprocessors but who wish they did. The step-by-step presentation does not require any computer or electronics background, so anyone who is interested can follow it. However, the book is not only intended for beginners. Engineers and technicians who are familiar with electronics will find the software descriptions valuable in updating their skills. Computer professionals will find the detailed treatment of the 8080 architecture and instruction set useful. Price $7.95.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to:
Kilobaud Book Department   Peterborough NH 03458
Be sure to include check or detailed credit card information

* *Add $1.00 shipping & handling charge for each order.* Note: Prices subject to change on books not published by 73 Magazine.

F-8 MICROCOMPUTER ON SINGLE BOARD

WITH F-8 CPU, FAIRBUG PSU, 3853 SMI-DEVELOPED FOR DESIGN ENGINEERS AND SERIOUS HOBBYISTS



F-8 MICROCOMPUTER Model 1080

Design Features Include:

• Buffered address and data bus to S-100 expansion connector • 4K program memory - 2708 sockets • 2K onboard page selectable static RAM • FAIRBUG monitor plus 1K 2708 custom monitor • Independent FAIRBUG scratch pad RAM • Memory protect switches • Buffered high speed data input port • I/O/interface selection (all 4 ports available) • Two sockets for I/O expansion • 20 ma or RS 232 interface • Quality solder masked PC board • Program control timers • Documentation • 64K addressing range • Prime parts

KIT: $249.00

Price postpaid Assembled $299
California residents add 6% tax
Send check, money order, corporate purchase order or COD orders to

Comptronics
19824 Ventura Blvd.
Woodland Hills, CA 91364
(213) 340-8843

C62

1977 KB BACK ISSUES

kilobaud — The Computer Hobbyist Magazine — ISSUE #1 — January 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #2 — February 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #3 — March 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #4 — April 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #5 — May 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #6 — June 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #7 — July 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #8 — August 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #9 — September 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #10 — October 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #11 — November 1977 — Articles — Features

kilobaud — The Small Computer Magazine — ISSUE #12 — December 1977 — Articles — Features

Issue #1—$5.00; #2 thru #12—$3.00 each.
Please send Kilobaud back issues:

___ issues JAN 77-$5	___ issues MAY 77—$3	___ issues SEPT 77—$3
___ issues FEB 77—$3	___ issues JUNE 77—$3	___ issues OCT 77—$3
___ issues MAR 77—$3	___ issues JUL 77—$3	___ issues NOV 77 —$3
___ issues APR 77—$3	___ issues AUG 77—$3	___ issues DEC 77—$3

TOTAL: ___________

Have credit card # handy and call
TOLL FREE (800) 258-5473

5/78 Kilobaud ● Peterborough NH 03458

Add $1.00 shipping & handling for each order.

☐ BankAmericard/Visa ☐ Master Charge ☐ American Express

Card # ______________________ Expiration date ________

Signature ______________________

Name ______________________

Address ______________________

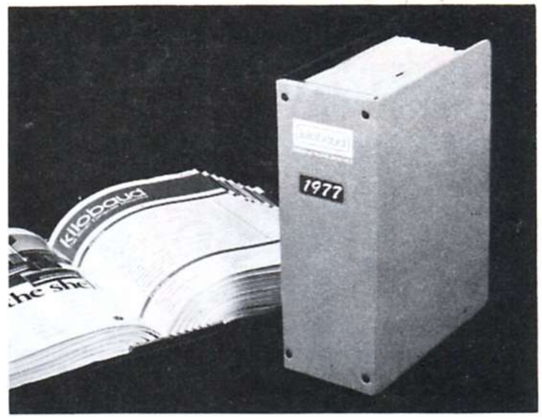
City ______________ State _____ Zip ________

COMPUTER MAILING LIST

Completely re-written and updated to include dealers, clubs, publications, and manufacturers. (It's the one we use for our mailings and we update it daily.) The list has over 2000 names painstakingly gathered from manufacturers, magazine ads, new product releases, hobby computer shows, and direct mail. You can buy this list printed on self-sticking labels for only $50.

kilobaud

PETERBOROUGH NH 03458



1977 kilobaud Binders

Keep your Kilobauds from being lost or damaged in these handsomely appointed binders with rich blue covers and gold lettering. Each binder holds 12 issues, making an EXCELLENT REFERENCE HANDBOOK. Several binders form a quality library you can be proud of.

Order yours today . . . only $6.50 each . . . 2 for $12.00.

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to:
Kilobaud Binders Peterborough NH 03458
Be sure to include check or detailed credit card information
Add $1.00 shipping & handling charge for each order.

SEND FOR THESE GREAT BOOKS TODAY!

● HOBBY COMPUTERS ARE HERE If you (or a friend) want to come up to speed on how computers work . . . hardware and software . . . this is an excellent book. It starts with the fundamentals and explains the circuits, the basics of programming, along with a couple of TVT construction projects, ASCII-Baudot, etc. This book has the highest recommendations as a teaching aid for newcomers. $4.95

● THE NEW HOBBY COMPUTERS! This book takes it from where "Hobby Computers Are Here" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards, a Baudot monitor/editor system, an audible logic probe for finding those tough problems, a ham's computer, a computer QSO machine . . . and much, much more! Everything of interest is there in one volume. Don't miss this tremendous value! Only $4.95

Have your credit card handy and call toll-free 800-258-5473, or use the card in the back of this magazine and send your check to:

KILOBAUD Book Dept. ● Peterborough, NH 03458

(Be sure to include $1.00 per order for shipping & handling.)

MOVING?

Let us know 8 weeks in advance so that you won't miss a single issue of *Kilobaud.*
Attach old label where indicated and print new address in space provided. Also include your mailing label whenever you write concerning your subscription. It helps us serve you promptly. Write to:

● KILOBAUD ● PETERBOROUGH NH 03458 ●

- ☐ Address change only
- ☐ Extend subscription
- ☐ Enter new subscription
- ☐ 1 year $15.00
- ☐ Payment enclosed (1 extra BONUS issue)
- ☐ Bill me later

If you have no label handy, print OLD address here.

AFFIX LABEL

Name ______________ Call ______

Address ______________

City ______ State ______ Zip ______

print NEW address here:

Name ______________ Call ______

Address ______________

City ______ State ______ Zip ______

This 8-bit machine, by itself, is as versatile as a lot of systems that include peripherals

Skeptical? For starters, because of its unique design the H8 is the only machine in its price class that offers full system integration, yet, with just 4K of memory and using only its "intelligent" front panel for I/O, may be operated completely without peripherals!

Memory Display

Register Display

I/O Port Display

In addition, by using the features of its built-in Pam-8 ROM panel control program, the H8 actually allows you to dig in and examine machine level circuitry.

Responding to simple instructions the "intelligent" panel displays memory and register contents, lets you inspect and alter them even during operation. And for greater understanding, the front panel permits you to execute programs a single instruction at a time. The result is a powerful, flexible learning tool that actually lets you "see" and confirm each detail of H8's inner workings.

If you need further evidence, consider the fact that H8's system orientation allows you an almost unlimited opportunity for growth.

Memory is fully expandable, the 8080A CPU extremely versatile, and with the addition of high speed serial and parallel interfacing you gain the added flexibility of I/O operation with tape, CRT consoles, paper tape reader/punches, and soon floppy disk systems!

The H8 offers superior documentation including complete step-by-step assembly and operation manuals, is backed by 54 years of Heath reliability, and comes complete with BASIC, assembler, editor, and debug software – others charge over $60 for!

H8, simplicity for the beginner, sophistication for the expert and at $375* just right for you.

*Prices are mail order net FOB, Benton Harbor, Michigan. Prices and specifications subject to change without notice.

HEATHKIT COMPUTERS

System Engineered for Personal Computing

H8

Be sure to use coupon on facing page of this magazine to order your FREE Heathkit Catalog!

INTERACTIVE COMPUTERS
The Microcomputer Store
7646½ Dashwood Houston, TX 77036
Phone 772-5257

Jim-pak electronic components

One-Stop Component Center

NEW!! EXPANDED PRODUCT LINE...

TTL
7400, 7402, 7404, 7408, 7410, 7420, 7430, 7432, 7442, 7447, 7473, 7474, 7475, 7476, 7485
7490, 7492, 7493, 74107, 74121, 74122, 74123, 74125 (DM8093), 74154, 74161 (DM9316), 74176 (DM8280), 74177 (DM8281), 74192, 74193, 74367 (DM8097)

Low Power TTL Schottky
74LS00, 74LS02, 74LS04, 74LS08, 74LS10, 74LS20, 74LS30, 74LS32
74LS73, 74LS74, 74LS75, 74LS83, 74LS85, 74LS86, 74LS90
74LS109, 74LS123, 74LS136, 74LS138, 74LS151, 74LS175, 74LS367

C/MOS
4000, 4001, 4010, 4011, 4013, 4016, 4017
4020, 4023, 4024, 4029, 4030, 4046, 4049
4050, 4051, 4069, 4071, 4081, 4511

Display LEDS Discrete
MAN2, DL704, DL707, DL747, DL750
XC556 Red, XC556 Green, XC556 Yellow, CLIPLITES—Red, Green, Yellow

LINEAR
LM301V, LM307V, LM309K, LM311V, LM320K-5 (7905), LM320T-5 (7905), LM320T-12(7912), LM320T-15(7915), LM339N, LM340T-5 (7805), LM340T-12(7812), LM340T-15(7815), LM555V, LM556N
LM567V, LM723N, LM739N, LM741V, LM747N, LM1458V (5558V), LM1488N, LM1489N, LM1556V, XR2206, CA3080, CA3130, CA3140, LM3900N (CA3401)

MICROPROCESSOR
Z-80, 8080A, 8212, 8224, 8228, 6800, 6810, 6830L8, AY-5-1013, 2513/2140, MM5314 (Clock Chip)
2101, 2102, 21L02, 7489, MM5262, 1702A, 82523, 2708, DM8835N, N8T97

SOCKETS
8 pin low profile, 14 pin low profile, 16 pin low profile, 24 pin low profile, 40 pin low profile, 14 pin plug, 16 pin plug
14 pin wire wrap, 16 pin wire wrap, 24 pin wire wrap, 40 pin wire wrap, TO-3 Socket, TO-5 Socket, Molex Pins

SWITCHES
Dipswitches, Toggle, Subminiature Push Button
Slide, Push Button

CAPACITORS
Aluminum Electrolytic
1mfd 50V, 4.7mfd 50V, 10mfd 50V, 22mfd 50V, 47mfd 50V
100mfd 50V, 220mfd 50V, 470mfd 50V, 1000mfd 25V, 2200mfd 16V

Ceramic Disc
10pf 50V, 47pf 50V, 100pf 50V, 220pf 50V, 330pf 50V, 470pf 50V
.001mfd 50V, .0047mfd 50V, .01mfd 50V, .022mfd 50V, .047mfd 50V, .1mfd 50V

Dipped Tantalum
.1mfd 35V, .22mfd 35V, .33mfd 35V, .47mfd 35V, .68mfd 35V, 1mfd 35V, 1.5mfd 35V
2.2mfd 25V, 3.3mfd 25V, 4.7mfd 25V, 6.8mfd 25V, 10mfd 25V, 15mfd 25V, 33mfd 25V

Polyester Mylar
.001mfd 100V, .0015mfd 100V, .0022mfd 100V, .0047mfd 100V, .01mfd 100V
.022mfd 100V, .047mfd 100V, .1mfd 100V, .22mfd 100V

CRYSTALS
CY1A, CY2A, CY12A

TEST CLIPS
14 pin clip, 16 pin clip

FUSE HOLDERS
HKP-3AG

P.C. POTENTIOMETERS
Single-Turn: 840P1K, 840P5K, 840P10K, 840P50K, 840P100K, 840P1meg
15-Turn: 830P1K, 830P5K, 830P10K, 830P50K, 830P100K, 830P1meg

DIODES
IN751, IN4733, IN4734, IN4742, IN4744
IN4148(IN914), IN4001, IN4004, IN4007, MDA-980-3

TRANSISTORS
C106B1, 2N2222A, 2N2907A, MJE2955, MJE3055
2N3055, 2N3904, 2N3906, 2N5129, 2N5139

CONNECTORS
DB25P Plug, DB25S Socket

DATA BOOKS*
7400/74LS Data Book
CMOS/Linear Data Book
Microprocessor/LED Data Book
*JIM-PAK Products only

HEAT SINKS

¼ W RESISTOR ASST.
¼ Watt 5%
50 Pc. Assortments

A new component product line of over 260 items developed for the independent dealer. Guaranteed nationally advertised products. Complete **JIM-PAK** program includes national advertising, direct mail programs, store display racks, stock rotation plan and return policy. For dealer information, write or call **JIM-PAK**, 1021 Howard Avenue, San Carlos, CA 94070 (415) 592-8097.

SEE JIM-PAK AT NEWCOM '78 — MAY 2, 3 and 4, 1978 AT THE LAS VEGAS HILTON

SEE YOUR LOCAL Jim-pak DEALER TODAY...

ALABAMA	
Mobile	Lafayette Radio Electronics
ARIZONA	
Fountain Hills	P & C Communications
Tempe	Computerworld Inc.
Yuma	Yuma Electronics
CALIFORNIA	
Bellflower	Earl's Hobby Shop
Berkeley	Al Lasher Electronics
Fontana	Fontana Electronics
Fullerton	Orvac Electronics Inc.
Long Beach	Scott Radio Supply
Mission Viejo	Tower Electronics Corp.
Monterey	Zackit
Oceanside	Electronic Center
Palo Alto	Zack Electronics
Pasadena	Dow Radio Inc.
Sacramento	The Radio Place
Sacramento	Zackit
San Carlos	J & H Outlet Store
San Diego	Radio Shack A.S.C. Mira Mesa
San Fernando	San Fernando Electronics
San Francisco	Zack Electronics
San Jose	Quement Electronics
San Luis Obispo	Mid-State Electronics
Santa Monica	Mission Control
Sunnyvale	Sunnyvale Electronics
Vallejo	Zackit
Walnut Creek	Byte Shop Computer Store
CANADA	
Alberta (Calgary)	The Computer Shop
Ontario (Willowdale)	Home Computer Centre
COLORADO	
Aurora	Com Co Electronics
Steamboat Springs	Norm's TV & Electronics
CONNECTICUT	
Bridgeport	Bridgeport Computer
FLORIDA	
Ft. Lauderdale	Computers For You
Lakeland	Lakeland Specialty Electronics
Orlando	Altair Computer Center of Orlando
Tampa	AMF Electronics
Tampa	Microcomputer Systems
FRANCE	
Paris	Computer Boutique
GEORGIA	
Atlanta	Atlanta Computer Mart
HAWAII	
Aiea	Delcoms Hawaii
Honolulu	Integrated Circuit Supply
IDAHO	
Idaho Falls	Audiotronics
ILLINOIS	
Evanston	Tri-State Electronics
Evanston	Itty Bitty Machine Co.
Groveland	Moyer Electronics
Mount Prospect	Tri-State Electronics
ILLINOIS (Continued)	
Oak Park	Spectronics
Schaumburg	Data Domain
INDIANA	
East Chicago	Acro Electronics
Hammond	Quantum Computer Works
IOWA	
Indianola	Electronix Limited
KANSAS	
Wichita	Amateur Radio Equipment Co.
KENTUCKY	
Lexington	Radio-Electronic Equipment Co.
LOUISIANA	
Baton Rouge	Davis Electronics Supply
MARYLAND	
Baltimore	Computer Workshop of Baltimore
Baltimore	Everything Electronic
Lavale	J & M Electronics
Rockville	Computer Workshop
Towson	Baynesville Electronics
Towson	Computers, Etc.
MASSACHUSETTS	
Medford	Tufts Electronics
North Adams	Electronics Supply Center
Waltham	Computer Mart Inc.
MICHIGAN	
Grand Rapids	Micro Computer World
Lansing	Fulton Radio Supply
Mt. Clemens	The Computer Store
MINNESOTA	
Duluth	Northwest Radio of Duluth
Eagan	Computer Room Inc.
MISSOURI	
El Dorado Springs	Beckman Electronics
Parkville	Computer Workshop of Kansas City
MONTANA	
Billings	Conley Radio Supply
NEBRASKA	
Lincoln	Altair Computer Center
Omaha	Omaha Computer Store
NEVADA	
Las Vegas	Century 23
NEW JERSEY	
Bayville	A.R.S. Communications Services
Cherry Hill	Computer Emporium
Hoboken	Hoboken Computer Works
Pompton Lakes	Computer Corner of New Jersey
Ramsey	Typetronic Computer Store
NEW YORK	
Albany	Fort Orange Electronics
New York	The Computer Stores Inc.
New York	Computer Mart of New York
Troy	Trojan Electronics
White Plains	The Computer Corner
NORTH CAROLINA	
Durham	Futureworld
Greensboro	Byte Shop
Raleigh	Byte Shop of Raleigh
OHIO	
Bucyrus	Mead Electronics
Cincinnati	Digital Design
Dayton	Altair Computer Center
Reynoldsburg	Universal Amateur Radio
Steubenville	Hosfelt Electronics
OKLAHOMA	
Guymon	Sound Service
Oklahoma City	Bits, Bytes & Micros
OREGON	
Beaverton	Altair Computer Center
Coos Bay	Herrick Electronics
Ontario	Miller Electronics
Salem	Computer Pathways
PANAMA	
Panama City	Sonitel, S.A.
PENNSYLVANIA	
Hershey	Microcomputer Systems Inc.
Murraysville	Computer Workshop of Pittsburgh
RHODE ISLAND	
Cranston	Jabbour Electronics City
Pawtucket	Jabbour Electronics City
SINGAPORE	Intertrade (PTE) Ltd.
SOUTH CAROLINA	
No. Charleston	Technical Services Inc.
TENNESSEE	
Clarksville	Masstronics
Knoxville	Byte Shop
Memphis	Sere-Rose & Spencer Electronics
Oak Ridge	Computer Denn
TEXAS	
Dallas	CompuShop
Houston	Altair Computer Center
Houston	CompuShop
Houston	Interactive Computers
San Antonio	Sherman Electronics Supply
VIRGINIA	
Alexandria	Computer Hardware Store
Alexandria	Computers Plus
Charlottesville	Lafayette Electronics
Richmond	Computers-To-Go
Springfield	Computer Workshop of North Virginia
Virginia Beach	Heathkit Electronics Center
WASHINGTON	
Bellevue	Altair Computer Center
Longview	Progress Electronics
Pasco	Riverview Electronics
Seattle	C-Com
Seattle	Empire Electronics
Spokane	Personal Computers
WEST VIRGINIA	
Morgantown	The Computer Corner
Morgantown	Electro Distributing Co.