The Small Computer Magazine

kilobaud

Understandable for beginners ... interesting for experts

5.00£

READY for BUSINESS

We've got it all together—the cost effectiveness and reliability of our 6800 computer system with a high capacity 1.2 megabyte floppy disk system. . . PLUS—an outstanding new DOS and file management system.

1 MEGABYTE DISK SYSTEM

DMAF1 introduces a new level of capability to small computer systems. This disk system features two standard size floppy disk drives using the new double sided disk and two heads per drive. Usable storage space of over 600 kilobytes per drive, giving a total of over 1.0 megabyte of storage on line at all times. Ideal for small business applications, or for personal "super" systems.

DMA CONTROLLER

The controller occupies one main memory slot in an SS-50 bus and uses the Motorola MC-6844 DMA controller. The combination of a DMA type controller and double sided disks give the system speed of data transfer unobtainable with smaller drives.

OPERATING SYSTEM

To compliment this outstanding hardware we are supplying equally superior software. The disk operating system and file management system is called FLEX. It is one of the most flexible and complete DOS's available for small systems, but just as important; it is easy to use.

No one can match the variety of compatible peripherals offered by Southwest Technical Products for the SS-50 bus and the 6800 computer system. Now more than ever there is no reason to settle for less.

DMAF1 Disk System (assembled) $2,095.00

DMAF1 Disk System (kit). $2,000.00

68/2 Computer with 40K of memory (assembled) $1,195.00

SOUTHWEST TECHNICAL PRODUCTS CORPORATION

216 W. RHAPSODY

SAN ANTONIO, TEXAS 78216
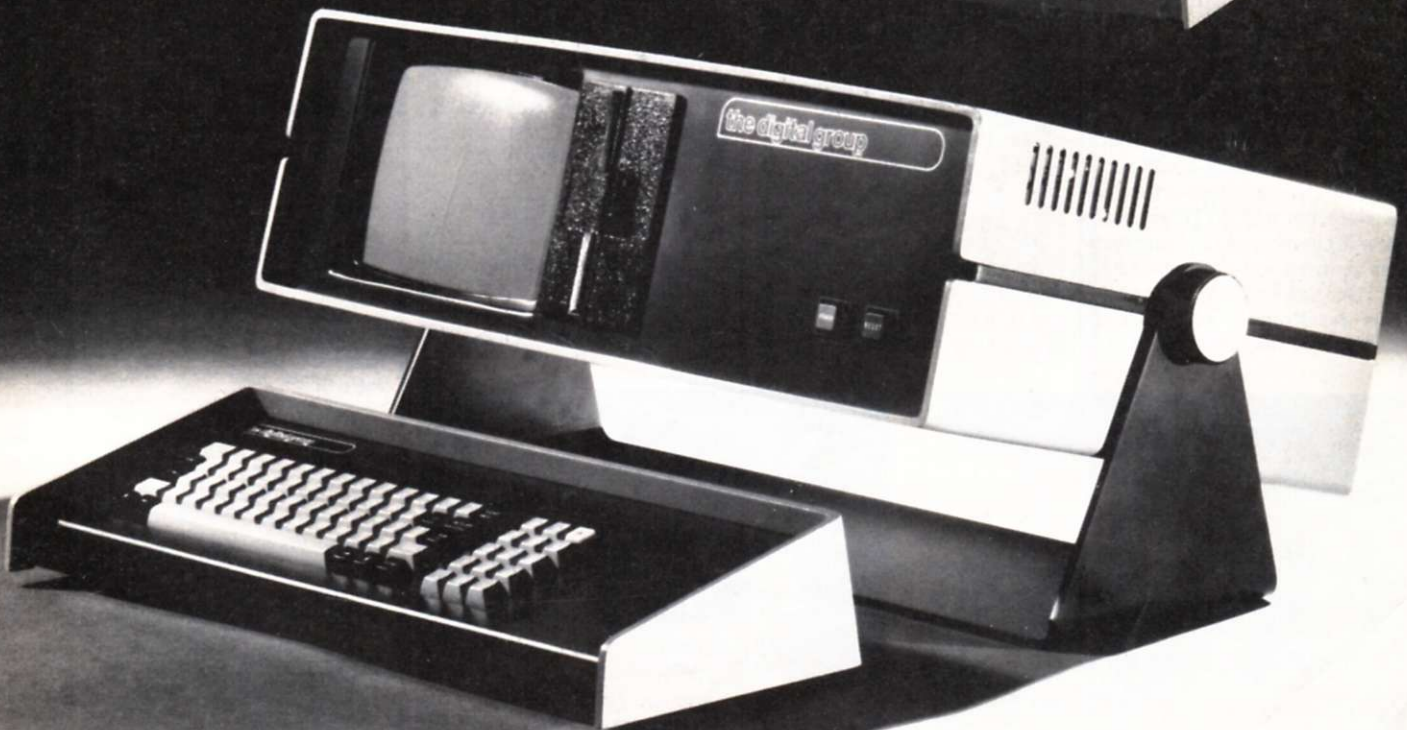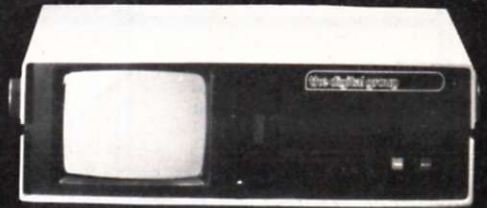
PUBLISHER'S REMARKS

Wayne Green

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

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 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

---

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*!

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.

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 Cybervi-

sion 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 con-

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

sists 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 sys-

tem 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 imag-

ination 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 Na-

tional 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 de-veloped 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.

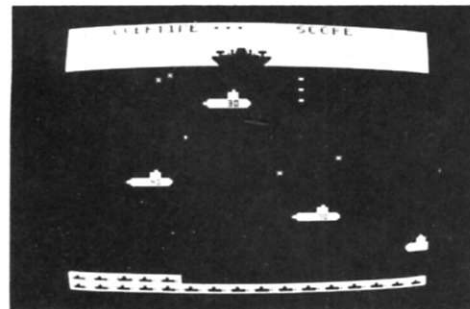*Photo 4. The employee lounge at Gremlin.*

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.

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 adminis-trator. 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.

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 let-

ters 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.

```
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.

Tom Kasper's Snoopy listing.

```
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
```

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° to 50° C, Storage 0° to 85° 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

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

NEW PRODUCTS

The Centronics S1.

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 Dig-ital 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.

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.

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.

TEI MCS-112.

The POP-1.

GRAPHICS
ARE
EASY

SO ARE HISTOGRAMS

```
1950    19.9
1960      25.4
1970        31.8
1980      27.6
1990      28.3
```
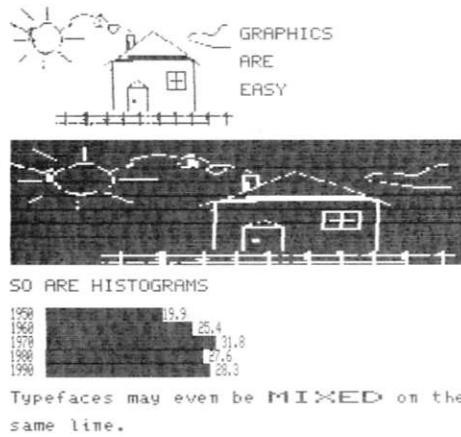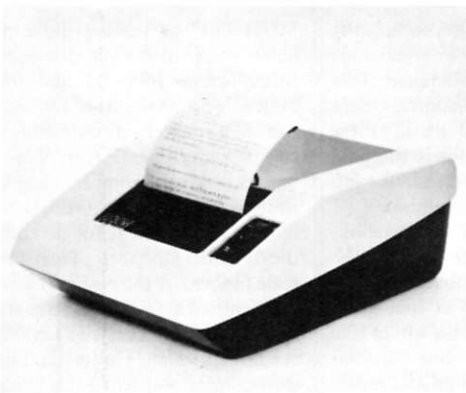
Typefaces may even be MIXED on the same line.
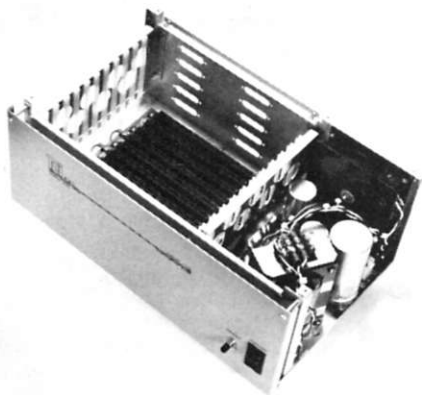
The Axiom 801 and sample printout from the 820 ($795).

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
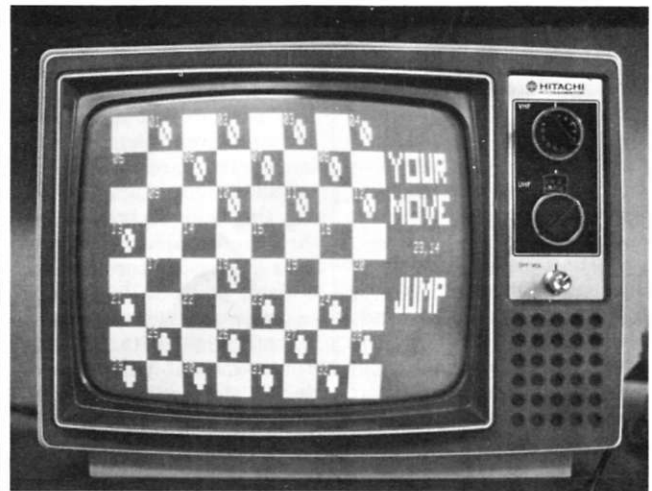
Video Checkers in action.

CLOAD Magazine.

North Star's ZPB.

BOOKS

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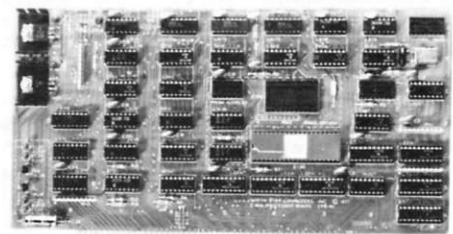 is that Osborne et al mean it; and what is even better, the book is for the potential user who doesn't know how a computer works, will probably never know and doesn't particularly care. It is written for a pure user orientation, treating the microcomputer as an appliance and recognizing that it is possible 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 ba- sically 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**

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 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, 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 $2\wedge(4096*8)$, 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
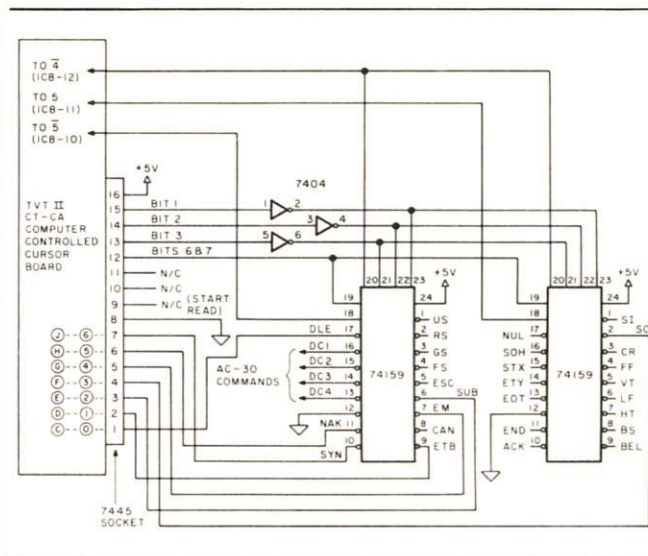
by the author. By leaving the TVT-II Cursor Control board in place, removing the 7445 and using the bit signals available at its socket plus the $\overline{4}$, 5 and $\overline{5}$ signals available elsewhere on the board, a 32-character, strobed decoder can be made using only two 74159s and half of 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 Compu-Craft micro APL program for the game of checkers with a Cyber-Clops 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.

---

computer 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

---

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 'he 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

```
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 micro-

---

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

---

```
   1          NAM    MUSIC1
   2 *
   3 * M6800 MUSIC PLAYING PROGRAM
   4 *
   5 * WRITTEN BY TERRY PERDUE
   6 *   KILOBAUD JANUARY 1978 P  94
   7 *
   8 * MODIFIED BY DON KORTE
   9 *   ALLOWS HALF REST, NOTE CODE=01
  10 *   ALLOWS WHOL REST, NOTE CODE=02
  11 *
0050          12          ORG    $50
0050 0900     13 SPEED    FDB    $0900          INPUT SPEED PARAMETER HERE
0052 0002     14 HSPD     RMB    2
0054 0001     15 TEMP     RMB    1
0055 0001     16 MUSIC    RMB    1              MUSIC TABLE BEGINS HERE
0000          17          ORG    0
0000 7F 0053  18          CLR    HSPD+1
0003 96 50    19          LDA A  SPEED
0005 47       20          ASR A
0006 97 52    21          STA A  HSPD
0008 7F 0005  22 REPLAY   CLR    $0005
000B 7C 0004  23          INC    $8004          CHANGE HARMONIC QUALITY BY
  24 *                                          BY ENABLING DIFFERENT
  25 *                                          OUTPUT BITS
000E 73 0005  26          COM    $0005
0011 8E 0054  27          LDS    #TEMP          POINT TO FIRST NOTE - 1
0014 DE 50    28 NUNOTE   LDX    SPEED          TIME PER NOTE
0016 33       29          PUL B                 PULL NEXT NOTE FROM TABLE
0017 5D       30          TST B                 LAST NOTE?
0018 27 EE    31          BEQ    REPLAY
001A C1 01    32          CMP B  #$01           HALF REST?
001C 27 14    33          BEQ    HREST
001E C1 02    34          CMP B  #$02           WHOLE REST?
0020 27 12    35          BEQ    WREST
0022 D7 54    36          STA B  TEMP
0024 4C       37 TONLUP   INC A                 REGA HOLDS THE VALUE WHICH
  38 *                                          IS SENT TO THE PIA
  39 *                                          COUNTING THESE AROUND GIVES
  40 *                                          HARMONIC QUALITY TO THE OUTPUT
0025 D6 54    41          LDA B  TEMP
0027 5A       42 COUNT    DEX                   COUNT DOWN NOTE TIME
0028 27 EA    43          BEQ    NUNOTE         DONE
002A 5A       44          DEC B
002B 26 FA    45          BNE    COUNT          AUDIO CYCLE NOT DONE YET
002D 87 8004  46          STA A  $8004          THIS CYCLE DONE
  47 *                                          STORE A NEW OUTPUT CODE
0030 20 F2    48          BRA    TONLUP
0032 DE 52    49 HREST    LDX    HSPD
0034 7F 8004  50 WREST    CLR    $8004          CANCEL OUTPUT FOR A WHILE
0037 01       51 WAIT     NOP
0038 01       52          NOP                   THEN WAIT
0039 01       53          NOP
003A 09       54          DEX
003B 27 D7    55          BEQ    NUNOTE         FINISHED
003D 20 F8    56          BRA    WAIT
  57          END
```

---

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 per-

sonal 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
```

21

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 an-nounced 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 at-tempt 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, Com-modore had never advertised the PET, but the magazine ar-ticles, television exposure and convention displays made it a pre-production marketing phe-nomenon. In fact, I assume the reader has already been ex-posed 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 ad-mission to "crummy documen-tation." 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 aug-mented by the published references and conversations with sales representatives.

To start with, I'll dive into the deep end of the pool of con-troversy 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 integra-tion* 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 poten-tial has not yet been adequate-ly 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, preced-ed by Jolt and KIM (see Photo 1). Although the Jolt is pro-duced by Microcomputer Asso-ciates, 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 hard-ware.

Of course, others, such as Dataworks, with its 5K of oper-ating 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, COUN-TER and I/O, was, in my opin-ion, almost as big a milestone in large-scale integration (LSI) progress as the microproces-sor 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

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

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.

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.
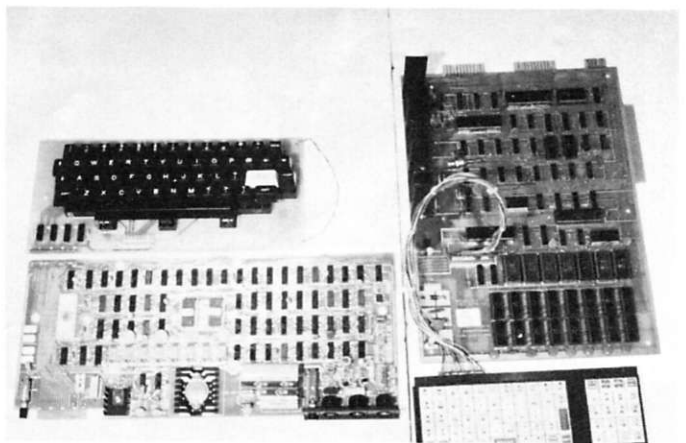

"PET gets As in three categories— vertical integration, good engineering and advanced technology."

---

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 PRO-LOG preceded KIM by a couple of years in adopting hex (hexadecimal), it cost nearly ten

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.

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

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

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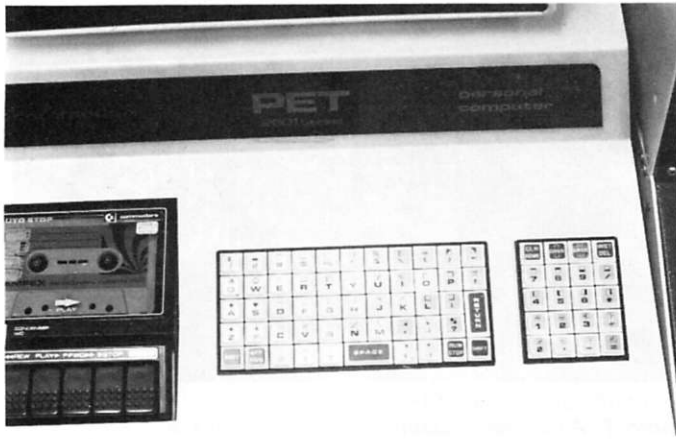
Photo 4. PET's controversial "calculator" keyboard, with quasi-standard key placement and conventional calculator number pad. Note variety of graphic symbols available with shift. Lowercase is also implemented (see text).

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.

25

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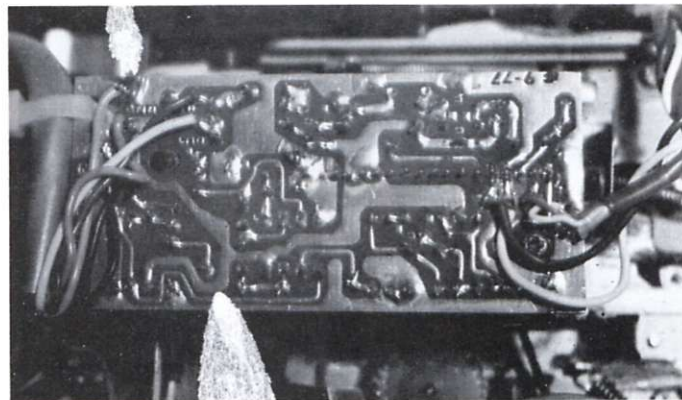
Photo 7. All of the cassette record, playback and erase electronics are on this single small PC card.

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 deci-

ters 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

phered, 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."

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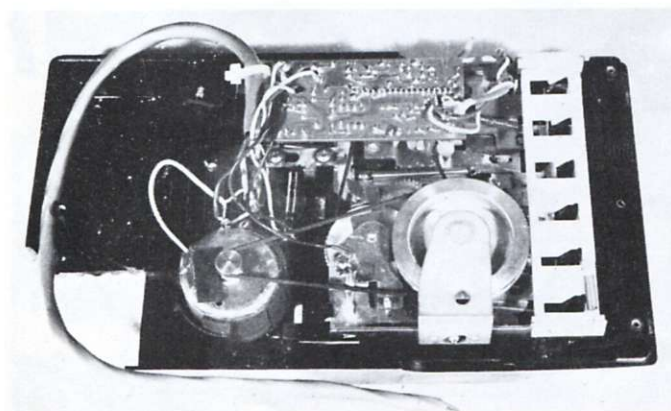
Photo 6. Gutted cassette is probably a stopgap measure. Note absence of usual electronics, speaker, jacks, etc.

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 for-

mat 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 mon-

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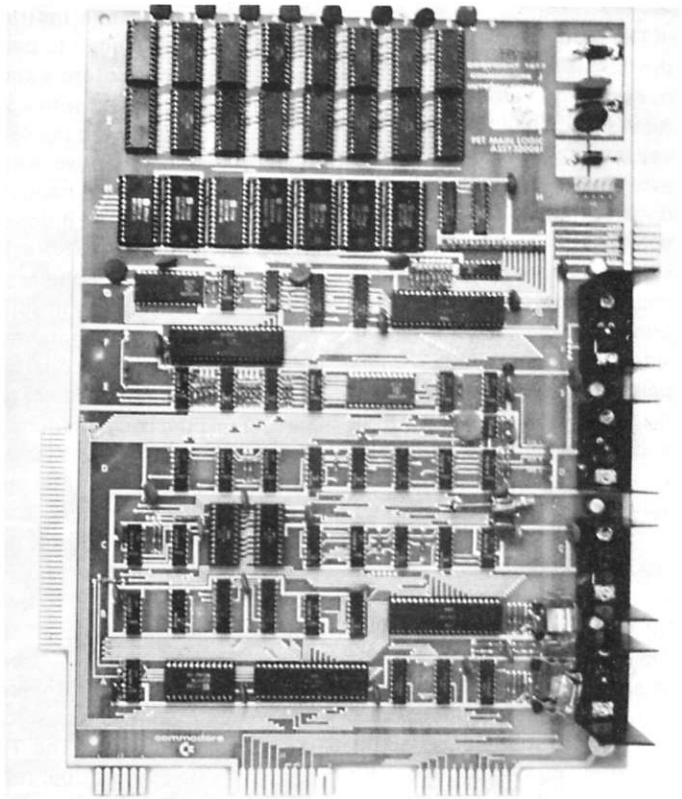
Photo 8. PET's brain: Top 16 chips are RAM. Seven ROM chips below contain operating firmware. Power supply and cassette #1 are along right side. Output ports are along bottom (rear). BUS and memory expansion are at left.

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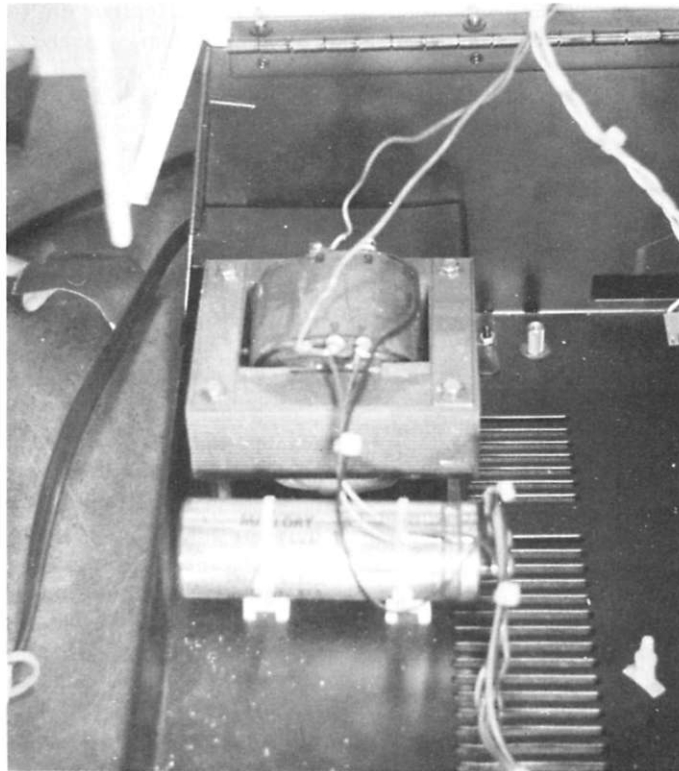
Photo 9. The power transformer, filter capacitor and 110 ac control are the only electrical devices directly wired to the chassis.

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.

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 ⚫f 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 de-

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

sign 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
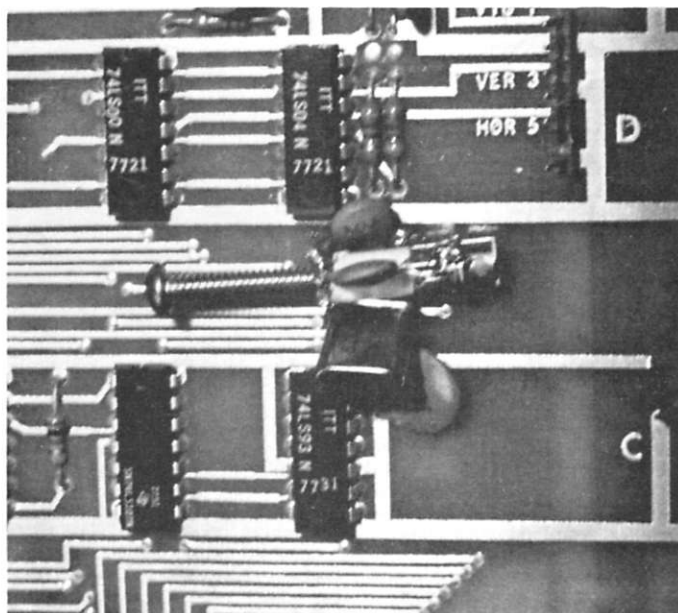
29

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

---

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

---

act as delimiters. If you're used to using abbreviated instruction, you'll be disappointed.

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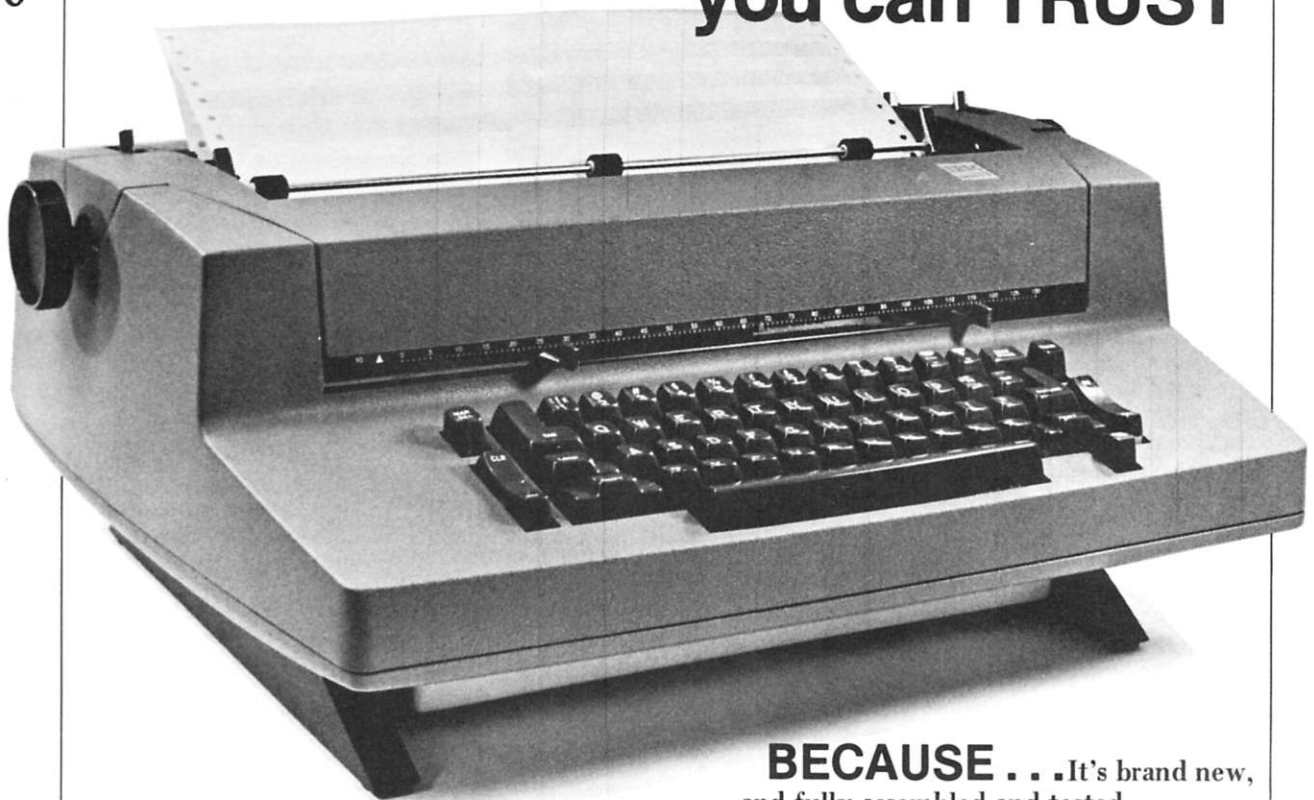
BECAUSE . . . It's brand new, and fully assembled and tested.

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 warranted 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.

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

SELECTERM

*Registered trademark of IBM Corporation   M30   *"Innovators to the Microcomputer Industry"*

31

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 manu-facturing 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 cam-switch 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!
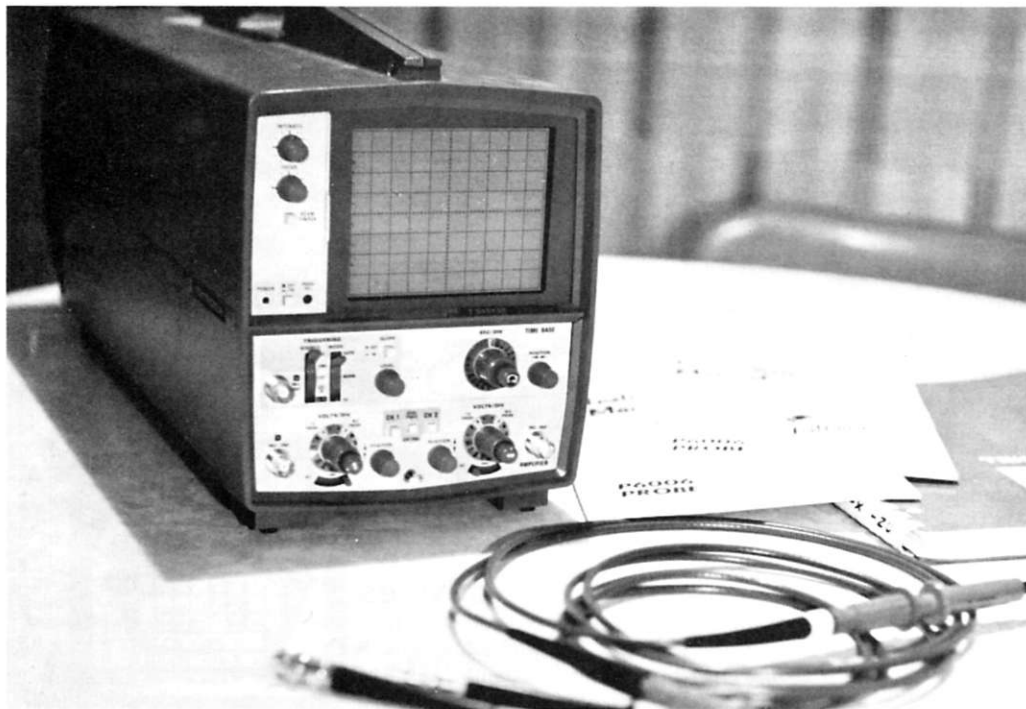
Photo 1. The complete system.
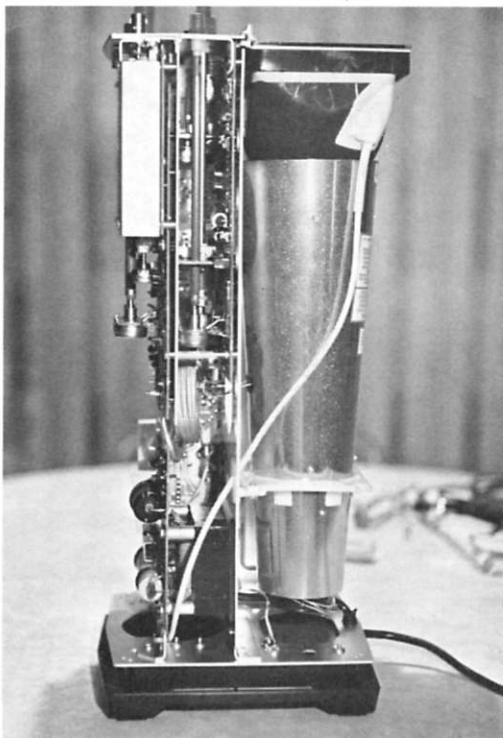
Photo 2. The tube determines the scope's length, width and height (right side view from front).
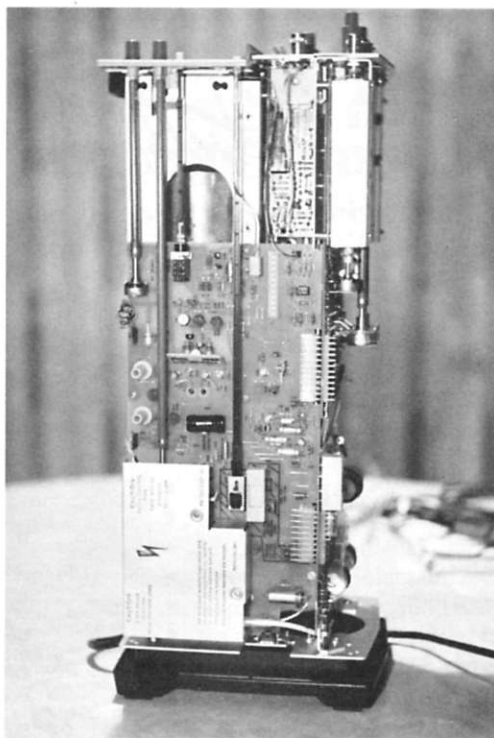
Photo 3. Most controls are extended (left side from front).

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. ∎

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

Photo 4. 8080 clocks 200 nanoseconds per division.

34

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.
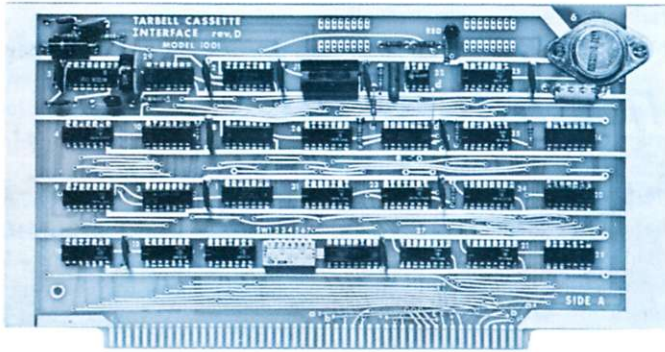

TARBELL CASSETTE INTERFACE

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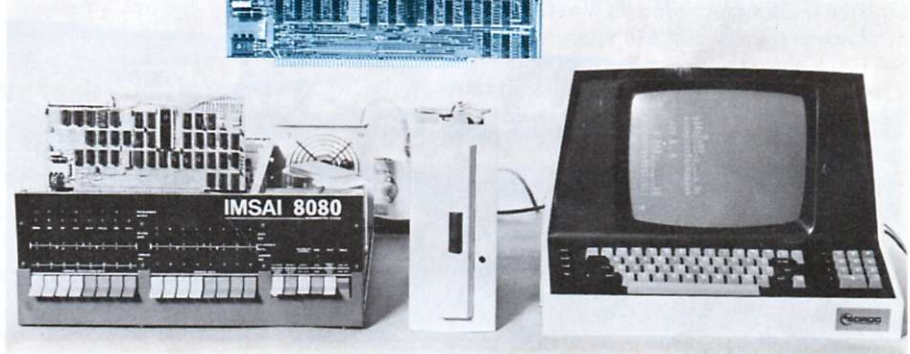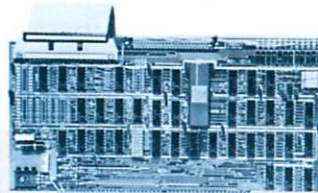
CP/M with BASIC-E and manuals: $100

Compatible Disc Drives

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

TARBELL PROTOTYPE BOARD
Model 1010
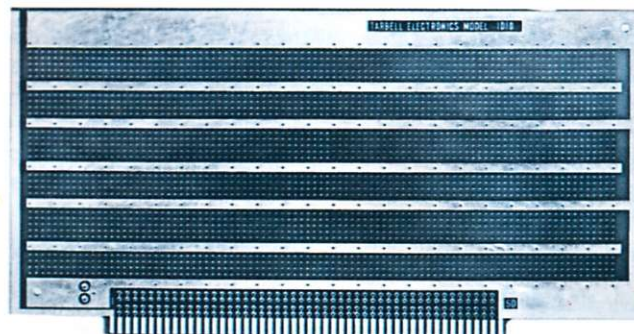
- Gold plated edge pins
- Takes 33 14-pin ICs or
- Mix 40-pin, 18-pin, 16-pin and 14-pin ICs
- Location for 5 volt regulator
- Suitable for solder and wire wrap
- ALTAIR/IMSAI compatible
  Price: $28.00

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

35

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 busi-

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.

ness. "Sometimes it is very hard to know what each sale actually costs."

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 screwups—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."

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

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."

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 soft-

ware 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.

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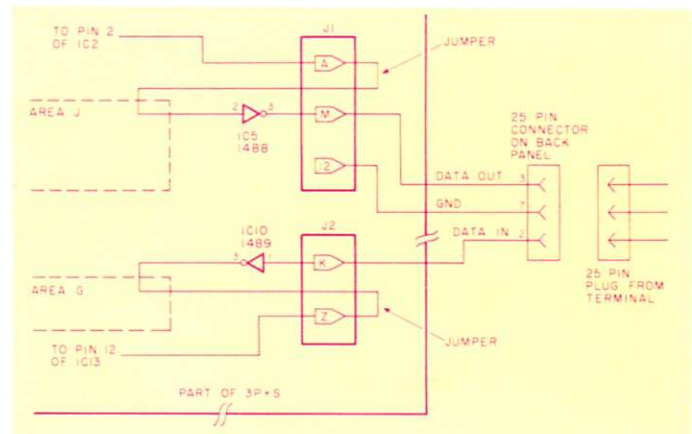
Fig. 1. The simple modifications to the Processor Technology 3P+S interface.

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 o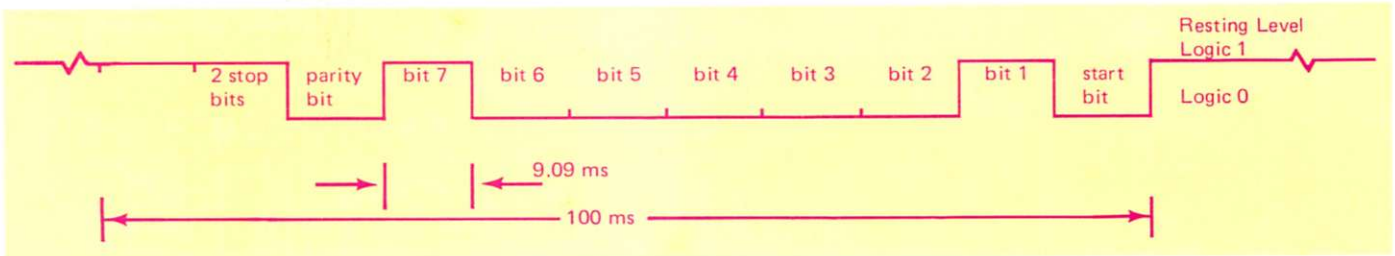f the circuit which is responsible for changing the UART's I/O signals to the appropriate drive level consistent with your particular peripheral.

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 CALI-BRATE. 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
	151	160		loop counter
VALU 3	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

MACHINE-LANGUAGE PROGRAMMING EXERCISES:

1. HOW MUCH IS 3A4 HEX IN DECIMAL?

SANDY

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.