Jan/Feb 1981 Volume 1, Number 1 $2.50 (USA) £1.20 (UK)

The magazine for Sinclair ZX80 users

SYNC

Converting From Other Basics

Games for the ZX80:
- Hurkle
- Acey Ducey
- Nicomacus
- Castle Doors

Reviews:
- 30 Programs for ZX80
- Linsac Game Packs

Tape Load Monitor for the ZX80

Two Methods of Producing Screen Graphics

Handling Pounds & Pence, Dollars & Cents

Directory of Resources

Glossary of Terms

LAMO-LEM PRESENTS:

CHEST OF CLASSICS

FOUR CLASSIC COMPUTER GAMES FOR THE MICROACE AND 4-K ROM ZX80.

LIFE
LUNAR LANDER
MINDMASTER
K-TREK

ALL ON ONE CASSETTE WITH GAME CHARTS AND GAME MANUAL INCLUDED **$9.95**

also: OLD #1
A CASSETTE OF FAMILIAR PROGRAMS: DICE, CHINESE RING PUZZLE, ETC., PLUS SOME NEW ONES. HERE'S YOUR CHANCE TO TRY OUR SOFTWARE ON YOUR MACHINE! **$2.99**

FREE!
WRITE FOR OUR FREE CATALOG & RECEIVE ZX80 & MICROACE CODING SHEETS.

LAMO-LEM, BOX 2382, LA JOLLA, CA 92038

SATISFACTION GUARANTEED . . .
OR YOUR MONEY BACK!

The magazine for Sinclair ZX80 users

SYNC

January/February 1981 Volume 1, Number 1

2 **Interview With Clive Sinclair** ..... *Ahl*
Past, present and future of Sinclair Research

4 **Sinclair ZX80** ..... *Tebbutt*
An in-depth review of the ZX80

8 **Building a MicroAce** ..... *McLaughlin*
The times, they are a changing

12 **Hurkle** ..... *McLaughlin*
Search for the beastie on a 10 x 10 grid

13 **Converting From Other Basics** ..... *Lubar*
FOR-NEXT loops

13 **Editorial** ..... *Lubar*

14 **Dealing Out a Conversion** ..... *Lubar*
How we converted a program

16 **Acey Ducey** ..... *Lubar*
The popular card game for the ZX80

18 **Nicomacus** ..... *Ahl & McLaughlin*
An ancient "boomerang" puzzle

19 **A Weekend With the ZX80** ..... *Singer*

20 **The SNYC Challenge** ..... *Stein*
Can you fit Hammurabi into the ZX80?

22 **Random Graphics** ..... *McGath*
Diagrams and pictures on the ZX80

24 **ZX80 Circuit Diagram and Specs** .....

26 **Crash Cursor—Origin** ..... *Truman*

30 **Castle Doors** ..... *Horton*
Cyclops, dragons, witches and zombies

32 **Draw a Picture** ..... *Branch*
Faces, dogs and flowers

34 **Dollars & Cents, Pounds & Pence** ..... *Gaunt*
Handling decimal numbers on the ZX80

38 **LED Load Monitor** ..... *Bridges*
A perfect load and read every time

39 **New Peripherals for ZX80** .....

39 **30 Programs for the ZX80—A Review** ..... *Lubar*

40 **Linsac's Game Packs—A Review** ..... *Stein*

42 **Resources for ZX80 and MicroAce** ..... *Staff*

45 **Sync Reader Survey** .....

46 **Glossary of Computer Terms** ..... *Staff*

Staff

Publisher/Editor-in-Chief	**David H. Ahl**
Editorial Director	**George Blank**
Managing Editor	**Jonathan A. Stein**
Associate Editor	**David Lubar**
Secretary	**Elizabeth Magin**
Production Manager	**Laura MacKenzie**
Typesetter	**Jean Ann Vokoun**
Financial Coordinator	**William L. Baumann**
Bookkepper	**Patricia Kennelly**
Customer Service	**Patricia Brown**
Order Processing	**Alan Kelly**
	Dorian Snipes
	Joan Swihart
Circulation	**Suzanne Guppy**
	Frances Miskovich
	Dorothy Staples

Index to Advertisers

Advertiser	*Page*
Computer Coin Games	23
Computer Music Record	29
Creative Computing	Cover 3
K. Macdonald	19
Lamo Lem	Cover 2
Linsac	7
Melbourne House	29
MicroAce	Cover 4
Sinclair	17
SYNC Magazine	44
SYNC T-Shirts	37

Volume 1, Number 1

SYNC is published bi-monthly by SYNC, 39 East Hanover Ave., Morris Plains, NJ 07950.

Subscriptions in USA: 6 issues $10; 12 issues $18; 18 issues $24. UK and foreign airmail subscriptions: 6 issues £10; 12 issues £18; 18 issues £25. Send subscription orders or change of address (P.O. Form 3575) to SYNC, 39 E. Hanover Ave., Morris Plains, NJ 07950. Call 800-631-8112 toll-free (in NJ, 201-540-0445) to order a subscription.

Controlled circulation pending at New Berlin, Wisconsin.

Copyright 1981 by Creative Computing. All rights reserved. Reproduction prohibited in any form. Printed in USA.

Interview with Clive Sinclair

David Ahl

Clive Sinclair is the man behind the first mass pocket calculator, the first scientific calculator kit and, now, a mass-market computer. I talked to Clive while in London at the PCW show.

Ahl: How did you get started in the electronics business?

Sinclair: I started in 1962 when I first came to Sinclair Radionics. We were in the kit business with hi fi systems and pocket calculators, (we made the first pocket calculator which is on display in the Museum of Modern Art in New York). From there we went into digital watches and TV sets and for a while we were linked with a government body in the U.K. At this point I left, as this arrangement did not work out too well. I took the same people with me and reformed the company as Sinclair Research Ltd. and launched the personal computer.

Ahl: Does Sinclair Radionics and its products still exist?

Sinclair: Only as a legal entity, but it doesn't trade.

Ahl: So effectively today your main product is ZX80?

Sinclair: That's right, but it's not our main research program. Our biggest research/development program is on flat screen television.

Ahl: Have you shown that at all? Or proved its capability?

Sinclair: Yes we have. We are at the stage where we have demonstrated it and will be doing so again later this year. We have pilot production and hope to shortly be at the next stage.

Ahl: What type of technology is used?

Sinclair: It is a flat screen Cathode Ray Tube.

Ahl: Does it have electron guns?

Sinclair: It has a conventional electron gun, but instead of the beam following a straight path, it bends to a right angle and it splits the screen.

Ahl: Is it monochrome or could it be color?

Sinclair: It is monochrome today, but eventually it will be color. We will pursue that.

Ahl: That sounds very interesting. Sharp had shown one about two years ago which provoked much interest in the electronics community but nothing ever came of it. Today it doesn't look like they are any closer to a commercial product than two years ago.

The philosophy of the company is very clear—to lead in technology.

Sinclair: What Sharp showed was an early development model of vertical projection, which, novel as it was, was a long way from a commercial product. But that is not the case with ours. Ours gives a picture which is not simply as good as a conventional Cathode Ray Tube, but it is better, and all the snags have been ironed out.

Ahl: Do you anticipate that the price of this will be competitive with the existing technology?

Sinclair: Absolutely. There is no way we would introduce something which cost more than existing technology. Not only does this have more feature advantages, it has price advantages over conventional technology. But in order to achieve this and because conventional tubes are made in such vast numbers, the only way it can be competitive is to have very large scale production. That is why it will take quite some time before the tube emerges.

Ahl: About the ZX80; how long has that been in development?

Sinclair: It started in March 1979 and it was put on the market in February 1980.

Ahl: How many people were involved? It seems to be quite a breakthrough on many fronts. Did you have simultaneous projects going on to bring it to fruition? Or was it mainly you, or a small team of people?

Sinclair: There were very few people involved really. I had the original idea and some of the system ideas. Then there were about two or three engineers who were involved in the detail and design; one engineer in particular did the final design and one wrote the software.

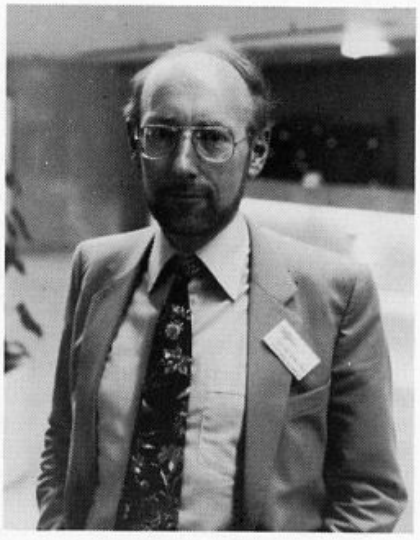
Ahl: Some people look at it and conclude that it is no more than a larger computer scaled down to smaller dimensions, while others will look at it and think of it as one of those language translators with a little more capability. From glancing at the specifications and seeing it at the Consumer Electronics Show, I have the impression that it has much more capability than that and that it does not seem to be just a scaled down computer. What is your concept? How do you view the machine?

Sinclair: While the ZX80 is a true computer in every sense, without any inherent limitations, it obviously was restricted in performance in its minimum configuration. We wanted to sell at as low a price as we possibly could, a computer upon which people could start to learn, really seriously, how to break into computers and how to really learn what computers were about. Now that might be an end in itself for many of them, but these may be executives who want to understand computing so when they buy computers for their firms or talk to people about the use of computers in their firms, they do so from a knowledgeable standpoint and they don't feel awed by it. Equally, we see it as a very powerful aid for students wanting to

---

Reprinted with permission from *Creative Computing*, December 1980.

learn computing. But at the same time it is expandable. We have just announced a Basic for it — which is more of a professional Basic than the original one and a 16K byte RAM pack for it at a very keen price, (around $100), so that it can be taken from the basic configuration to a really very powerful system at a very low cost. Hence our price will be somewhere around half the conventional competitions. I would make the point that in order to do this we have done everything in-house, i.e., in the way of anything having to be specially done we had to write our own Basic because we wanted to achieve performance in the Basic as it wasn't available on the existing Basics.



Ahl: Is that Basic compatible with Micro-Soft Basic or some other Basic on the market or has it got a lot of bells and whistles that make it incompatible with others?

Sinclair: It is not compatible. It is our own. We had to take that step in order to achieve the bells and whistles that you mentioned and have those features that we feel are important to our customers. In order to hit our cost goal we had to do a better packing job than had been done before. The ROM in our basic machine is just 4K bytes which contains everything — Basic, operating system, keyboard control and display I/O. Now there is no way we could have done all that with an off-the-shelf Basic. Furthermore, we then pack data into the RAM at least four times as tightly as anyone else. We would not have been able to do that unless we had our own Basic. Another thing that we wanted was keyword entries. You may have noticed that by pressing a certain key a keyword is entered. So there is very little for a non-typist to do. A lot of people are put off in the learning stages with having to type PRINT every time and abbreviations can be misleading. We felt this was very crucial and it is not included in standard Basics, so we had to choose to go away from the existing patterns.

Ahl: Pertaining to the marketing of the ZX80, do you anticipate selling it through computer stores, department stores, or other types of outlets or through direct-mail as you have done with some of your other products?

Sinclair: We are going to sell the computer by direct-mail in England and the U.S. There may come a time when store distribution is possible, but you can't really distribute to a store until you have a much better informed public. A very large mail-order campaign may help do that, so it may be possible later to distribute to stores.

Ahl: Many people buying computers today, particularly at this price level, are concerned with where or who can they go back to for service, particularly if they buy a machine by direct-mail.

Sinclair: There are two points there: the first involves the product, the second is who to come back to for service. We have a permanent office in the States and we stand behind our product at all times. We haven't seen anything but 1% or under 1% failure rate in the field in the U.K. It is a very reliable product so we haven't got a serious problem. We operate with service contract houses and this is all part of the guarantee of the product.

Ahl: Jumping back to the technical aspect of the ZX80, it appears that although now that it is designed for a power supply to be plugged into standard power, it could probably be battery operated and completely portable. Is this in your plans?

Sinclair: Yes it is. It has a very low power consumption and could certainly run for a reasonable time on batteries. Of course you need a battery T.V. set as well.

Ahl: What about other peripherals, floppy disk, mass storage, printers etc.? Are they in the works?

Sinclair: Yes they are, we have a floppy disk coming in about a month; the other items should be out the middle of next year.

Ahl: How do you feel about other alternative forms of mass storage? Some people have said that the bubble memory is coming down in price fast enough that it may in two or three years replace the floppy disk as a mass storage device. Do you think that's likely to happen?

Sinclair: The price projections we see from manufacturers over the next two or three years don't suggest that to me.

Ahl: What else would you like the world to know about you, the product, philosophy etc?

Sinclair: The philosophy of the company is very clear — to lead in technology. We've got a good start in this field and we plan to take as good a lead as we can. Some of the things we are developing, such as the flat TV screen, a truly portable system, etc. are coming along too. Clearly we think these are things we can do. □

David Tebbutt

sinclair ZX80

Clive Sinclair has surprised the world with the launch of his attractive, hand-held personal computer. Costing less than $200, and plugging into a TV and cassette recorder, some now suggest the ZX80 to represent the thin edge of a mass consumer market wedge.

Introduction

Sinclair Research set out to build a simple to use personal computer running Basic and capable of breaking the psychological price barrier of $200. Well, they succeeded with their ZX80. Why ZX80? No reason really except that it's based on an NEC copy of the Z80 processor chip . . . and it sounds nice.

The machine is available by mail order only; there are no plans to sell it in stores — yet. The kit version is only available in the U.K.; in the U.S. the assembled version costs $200 and includes an AC power supply.

The ZX80 is amazingly light, twelve ounces in fact, and easily held in one hand. The low weight is achieved through use of a moulded plastic casing just 1mm thick.

It connects quite happily to the television set and the cassette recorder, although it might take a few minutes to find the optimal settings. Once attached to the TV, it gives a rock steady display (more on that later).

Hardware

I have to say that I think it very pretty (our art director would probably beg to differ) — the casing even has go-faster stripes, which look suspiciously like ventilation slots in black and white photographs (coincidence, I'm sure). I would, however, have been happier with something rather more sturdy; ABS plastic might have done the trick, although perhaps at the cost of attractiveness.

The keyboard is most interesting; it's one of those waterproof, chemical proof, completely sealed units and it's stuck on to the main printed circuit board (PCB). Made of a special tough plastic, the undersurface is printed with the key symbols so as to eliminate any rubbing off. Between this keyplate and the PCB containing the metal contact strips (about five per key) is a piece of sticky plastic containing forty holes which line up with the "keys". This material is about .006" thick and is just sufficient to keep the metal underside of the keyplate away from the contacts, except when touched of course.

Typing gives a sensation of drumming your fingers rather than of doing anything useful. This is a totally mistaken impression because it really works rather well. For those who are interested, I found that a "wiping" action was more successful than the tapping movement usually associated with typing. Typists may be pleased to hear that the keys are in standard QWERTY layout although somewhat compressed compared to, say, the office IBM.

Looking inside the machine, I find that it's controlled by an NEC 780-1 processor chip . . . a copy of the well known and very successful Z80. This CPU, running at 3.25MHz, does all the work for the ZX80, including driving the TV and the cassette recorder. You'll notice that if any work is taking place, be it calculation, accepting input from the keyboard or driving the cassette, then the TV picture disappears — only to return when the activity is complete. This can be irritating to observers (at a demonstration for example) but I found it positively beneficial when keying in programs because it gave me positive feedback whenever a key made successful contact.

This article has been reprinted with the permission of *Personal Computer World*, a British publication covering the personal computer field. Since we cannot make frequent trips to England we keep aware of the latest British developments through magazines such as *Personal Computer World*.

A one-year subscription to *PCW* costs £8 in the UK or £22 in the USA (via air). Write Personal Computer World, 14 Rathbone Place, London W1P 1DE, England.

The Basic interpreter, operating system, character set and editor are all held in a 4K Byte ROM. If you are feeling adventurous there's no reason why you shouldn't pop your own ROM (TMS 2532) in its place.

Memory in the basic system comprises 1K static RAM; you can add to it via the expansion port, giving a maximum potential of 16K. The memory expands with the aid of plug in modules, each designed to carry up to 3K in 1K increments. Thus five modules would be required to give the 16K maximum. At switch-on the machine does a memory check which also tells the system how much memory is on-line. Should you reconfigure the memory, then the command NEW will execute the memory check cycle again.

Moving on to the "outside world" connections, there's a cassette interface, TV socket and a hefty edge connector. The cassette interface comprises two 3.5mm jack plug sockets, securely mounted on the main (and only) PCB. One connects to the "ear" socket on the cassette recorder and the other to the "mic" socket. There is no facility for remote control of the cassette motor.

Although I encountered one or two problems at first, once working, the cassette interface proved trouble free. My particular recorder had a nasty habit of recording noises when the CPU was "sending out" silence. This caused the system to get its knickers in a twist when reading from cassette because it expected silence just before the file header record. After a couple of hours (what a confession) the culprit was found — the "ear" lead, which acts as a monitor while recording, was setting up some sort of oscillation. Answer — simple — disconnect the "ear" jack when recording. Another tip which ensures trouble free loading is to move the tape into the silent section before issuing your LOAD instruction. Rumor has it that the cassette operates at around 250 baud — I believe it, although it doesn't seem terribly important when you're only loading the 1K that I was.

The television connector is simplicity itself. Plug one end of the cable (supplied) into the ZX80 and the other into the television aerial socket, tune to channel 2 and you're in business. The display is magic; rock steady and very clear although reversed characters (white on black) are not so good.

I have already mentioned the business of the display switching off every time the processor needs to do something else. If this drives you mad then you'll have to forfeit some of the undoubted pleasures that this machine has to offer. The screen is not memory mapped; it's treated like a serial file — like a printer in fact — which means that fast moving graphics are out of the question. No doubt some clever Dick out there will take up the challenge and fudge the system, just to prove me wrong. More about the reasons for this in the Software section, but anyone who is hooked on white characters on a black background can suitably modify the PCB, though why they should want to I'll never know. It's a matter of cutting one track and making a small bridge to another.

Do you take your computer camping with you? You'll be pleased to hear that it can run from a car battery, provided that the lead regulates the supply. I believe you can buy a cigarette lighter plug with a built in regulator . . . couple that with a portable TV and a battery powered cassette recorder and you'll be the envy of the campsite.

Technical Specification

CPU:	NEC 780C-1 (copy of Z80) 3.25 MHz
Memory:	1K static RAM, expandable to 16K
Keyboard:	Keyplate, under-surface printed
Screen:	Use own television. Pixel graphics 24 lines x 32 chars.
Cassette:	Use domestic audio cassette recorder.
Bus:	Edge connector with 44 lines — 37 from CPU, 0V, 5V, 9V, Clock, External memory indicator and two earths.
Software:	4K ROM containing Basic, Editor and Operating System

The keyboard is most interesting; it's one of those waterproof, chemical proof, completely sealed units.

Now let's look at the hefty edge connector. This is where the memory expansion modules fit in, each one being "piggy backed" on the one previous. Thus there are always 44 contacts available for outside use. There are 37 lines drawn from the CPU plus 3 power lines (at 0V, 5V and 9V); the other lines comprise two grounds, a "clock" signal and an "external memory in use" indicator.

All in all, the Sinclair ZX80 is a well designed, well produced personal computer. Memory addition comes a bit expensive at about $700 for the full expansion but Clive Sinclair tells me bigger RAMs are on the way — that means cheaper expansion when they appear.

I'm sorry that there are no pictures of the machine's innards. The fact of the matter is that I was given one of the development machines which had a couple of "Veroboarded" EPROMS and a selector IC floating around on the ends of some pieces of wire which in turn were soldered into the "official" ROM socket. I thought it best to spare Mr. Sinclair's blushes.

Software

The software of the ZX80 comprises the Basic interpreter, the Editor and whatever it is that does the rest of the work (Operating System seems too grand a title). Rather than looking at each separately, I shall examine them in the order they might be encountered.

First of all the keying in of programs. For two reasons it's an absolute joy! First you don't have to type in many of the Basic instruction codes, one key is sufficient; second you cannot enter anything that is syntactically incorrect. Some Basic instructions have to be entered the long way (these are listed above te keyboard) but 29 of the instructions may be entered with a single keystroke, while only 8 need to be keyed in full.

As with many small computers most of the instruction codes are stored in a single byte. Normal Z80 machine code can be entered using the POKE statement and executed with the USR instruction. This should keep the buffs happy after they have tired of Basic. Syntax checking is superb — it's impossible to go wrong. Every character is checked on entry and, if the interpreter thinks that you are going to make a mistake, it signals with a reverse S (for Syntax) at the point it thinks you have gone wrong. If, later in the same line, you correct the error, then the marker disappears. What a grown up facility for such a small machine! Incidentally, the program lines are displayed very clearly with line numbers, instructions, operators and what have you being nicely spaced out.

Inside the memory, however, there's a completely different story. The lines of code are held as compactly as possible with most of the commands and operators occupying one byte each. The spaces are removed and there are very few extra bytes needed — for instance the new line code is one byte, although I did notice that the "=" operator needed one extra for some reason. I'm sure there are others, but I'm equally sure they are very few and far between. An example of the storage requirement is as follows:

10 FOR A = 16424 TO 17424	18 Bytes
20 PRINT PEEK(A);	12 Bytes
30 NEXT A	5 Bytes
40 STOP	4 Bytes

So you see, the storage for that program (displaying the 1K memory) is 39 bytes long — an average of 10 bytes per instruction. I'll leave you to work out what sort of program you can get in 1K. Perhaps I should mention that the screen buffer uses part of the 1K, as does the stack and system control area. The stack is held at the top of memory and "grows" down; I put 327 entries on it before it stopped accepting them.

The program and variables "grow" up into the screen buffer thus reducing the amount of data on display. Eventually it's possible for the program or variables to grow so large that there's nothing left on display. It was while experimenting with this interesting feature that I crashed the system. It seems the software couldn't cope with someone entering a string 868 bytes long! After about 424 bytes of input the screen removed another character every time I keyed in a new one — it was most odd to watch.

Another way of crashing the system, in fact the only other way I could find, is to hit the EDIT key while in the middle of an INPUT loop. This returns the current program line with a syntax error which is impossible to clear. For those who are feeling unhappy about all this talk of crashing systems, don't worry, it's not as bad as it sounds. In the first place you have to enter forty characters after the screen has gone blank, and in the second place you can only hit EDIT when you are also holding the SHIFT key down.

Now it may be that, having loaded your program, you wish to edit it. Well once again there is some rather excellent software to help you. The Editor enables you to move a "current line marker" up and down the program text. Wherever it is you will always be able to see the marked line and at least some of its neighbors (it's called getting it in context). Pressing the HOME key causes the marker to disappear — it has in fact gone to an imaginary position, one above the first program line. Having reached the line to be edited press the EDIT key and the line will be presented at the bottom of the screen ready for you to do your worst. From now on it is as if you are entering the line for the first time.

The benchmark* timings show the ZX-80 to be very fast, even though I had to introduce some extra code to make some of the instructions work. Specifically I had to bracket expressions like LET A=(K/K)*K)+K—K . . . if I hadn't, the expression would have exceeded the ZX80's capacity. The machine can only operate on integers and these must have values from —32768 to 32767. I couldn't execute Benchmark 8 because the machine has no logarithmic or trigonometrical functions built in.

*PCW uses 8 Benchmark programs to compare computers.

Basic

String Expressions			
CHR$(n)	TL$(s)	STR$(n)	
Integer Expressions			
PEEK(n)	CODE(s)	RND(n)	USR(n)
ABS(n)			
Statements			
NEW	LOAD	SAVE	RUN n
RUN	CONTINUE	REM	IF n THEN stmnt
INPUT dest	PRINT	LIST n	LIST
STOP	DIM (n)	FOR =n TO n	GOTO n
POKEn,n	RANDOMIZE n	RANDOMIZE	CLEAR
CLS	GOSUB n	RETURN	NEXT
Operations			
n**n	— n	n*n	n/n
n+n	n — n	n = n	n > n
n < n	s = s	s > s	s < s
NOT n	n AND n	n OR n	

n = number
s = string
** = to the power of

Finally, it's possible to save programs and any variables associated with them. If you want to make use of those same variables when reloading the program, use GOTO rather than RUN. Although it's possible to SAVE programs in this way, no provision has been made to save files — yet.

Every character is checked on entry.

That's about it for the software; once again, considering the size of machine and price, I think that it's not at all bad.

Basic

The ZX80 Basic has been well thought-out and, while it lacks some of the elegance and sophistication of the bigger machines, it's a very usable version of the language.

The main limitations relate to file handling and mathematical functions. File handling facilities don't exist, except by SAVEing the whole of memory (which is probably not as daft as it sounds). It does mean that you can save a program with all its variables, reload it the next day, remember to kick off with a GOTO rather than RUN, and carry on from where you left off. On the small memory machine it doesn't seem that important, but on the larger memory machines it means you can hold some reasonable sized files together with your program.

Benchmark timings (in seconds)

BM1	1.46
BM2	4.69
BM3	9.18
BM4	8.95
BM5	12.7
BM6	25.9
BM7	39.2
BM8 not performed (see text)	

The mathematical limitations are possibly more serious. The fact is the Basic can only handle integers in the range — 32768 to 32767, no decimals, hence the programmer must write a little routine for each mathematical function that requires decimals to be used. This should pose few problems for those with the larger memory machines but it will undoubtedly occupy a fair chunk of the basic 1K system.

So much for bad news; now here are some of the good features of the language.

Taking numeric functions first, the Basic offers up to 26 single dimension numeric arrays of any length. It also allows three Boolean operations — AND, OR and NOT.

The randomizing functions are worth a mention. RANDOMIZE n sets a seed value, while RND(n) gives a random number in the range 1 to n. PEEK and POKE are both available so it's possible to read or modify memory contents; coupled with the USR function, this means that Z80 machine code routines can be executed.

Up to 26 FOR . . . NEXT loops can be nested and the number of nestable subroutine calls seem to be dictated by the amount of memory available to the stack. On the 1K machine with a short (4 line) program, I ws able to get 327 subroutine calls in before needing to RETURN.

String functions, while adequate, could definitely be improved. The absence of a DATA statement and the lack of string arrays caused particular frustration. Although there are ways around these problems, they can be time consuming and messy.

The functions which are available, and which form the building blocks of string handling subroutines, are STR$, TL$, CODE, CHR$ and INPUT. STR$(n) returns a string of 1 to 6 characters representing the signed, decimal value of n. TL$ returns a string minus its first character, while CODE returns the code for the first

character in a string. CHR$(n) represents the character whose value is n and INPUT allows the operator to input numeric or alphanumeric information. A nice touch is that if the destination of input is a string variable, then the Basic kindly provides a pair of quotes which act not only as a prompt, they also save a little bit of keying.

There is one trap here for the unwary, and I fell into it. I had this nice little loop going and after a while I got fed up with it. Could I get out of the system — could I heck! I hit everything in sight but all I managed to do was crash the system (see earlier). The trick is that if you are in an input string loop, remove the quotes and then put in an arithmetic expression which will resolve outside the range — 32768 to 32767.

So, that's the Basic — I reckon that it's pretty good under the circumstances and in some respects I prefer it to the Basics that do all your thinking for you.

Documentation

This comprises a programming cum operating manual. It's very well presented, being written by Hugo Davenport of Cambridge Consultants, with appendices by the mystery man from Cambridge who wrote the Basic interpreter. There are a few small mistakes in the manual — none of them terribly serious and all of them being dealt with before the next reprint. It's probably good enough to learn to program from it and my only real criticisms like in the area of what it does (or rather, doesn't do) for the raw beginner. I lent the machine to one such person for a few hours and here in his reply regarding the documentation:

"I read Chapter 2 (Getting Started) and got completely lost by the third page. One minute it's telling me how to wire everything up, the next there's something incomprehensible about storing programs on tape. I couldn't find an 'Idiot's Guide to getting started' anywhere."

Maybe the Operating Manual wasn't designed with such a person in mind — even so, novices like him must surely represent a good sized chunk of the ZX80's ownership potential."

Future Plans

A new ROM is being developed which will overcome most of the shortcomings of the existing system. Being 8K instead of 4K means that file handling routines will enable us to read and write tapes — even discs! This new ROM will also include the missing trigonometrical, logarithmic and floating point arithmetic functions.

Another area of development is on the memory front. A 16K plug in dynamic RAM is a distinct possibility; this will be considerably cheaper than taking the present $700 expansion route. A printer is also likely to appear in due course.

Potential Use

In its present form the ZX80 offers an ideal introduction to computing. It makes Basic easy to learn, it's small enough for it not to be intimidating and it's cheap enough that, should you decide computing is not for you, you can give it away, sell it or whatever. Indeed it's probably cheaper to learn Basic this way than to pay for many of the courses around.

Teachers might buy it for their students' use because at the price there is no need to go through a complicated rigmarole to get the money. The 1K version can be used for fairly simple games and activities, although it's likely you will want to expand it before very long. Later, when the file handling facilities are introduced together with floating point arithmetic, I think the machine will become really useful, though still very much at the personal level. Home accounts and engineering calculations spring to mind immediately — don't ask me why! Suddenly the machine becomes something more than a teaching machine or toy: it starts to become a real computer.

Conclusion

Having just read Science of Cambridge's claims for the machine again, I have to say I agree with most of them. The only point I would question is that it offers high resolution graphics. OK, OK, so they are playing the same game as everyone else . . . all the same I feel that it should be explained. Just lately, people have taken to calling pixel graphics, high resolution graphics. Accordingly, what used to be called high resolution graphics now has to be called ultra high resolution graphics. To put it another way, the ZX80 offers a graphics resolution of one quarter of one character, plus you must write your own software to be able to use it. PET is in exactly the same boat, unless you want to buy the high resolution add-on at about $600.

The ZX80 appears to be a well thought out machine both in terms of hardware and software. It has an excellent editor and interpreter which between them help you avoid all sorts of nasty pitfalls. The Basic instruction set lacks one or two fairly important facilities — namely file handling and floating point calculations. Despite this, it's still a fine machine on which to learn about computing. The new ROM expected later this year will overcome the prime limitations leaving me very little to say except that I hope Mr. Sinclair and his merry men of Cambridge can cope with the expected flood of orders and, perhaps more importantly, the after sales service which is vital in this sort of operation. □

Our thanks go to Clive Sinclair for lending us the machine, and to Jim Westwood (its designer) for patiently answering so many questions.

LINSAC

68 Barker Road
Middlesbrough
TS5 5ES

THE ZX80 COMPANION
(Second Edition)

ISBN 0 907211 00 3. Price £10.00 incl. UK postage.

This best-selling manual on the Sinclair ZX80 covers ZX80 BASIC, hardware and programs and has a detailed explanation of the ZX80 Monitor, routines and entry points.

Ch.1 - Operating the ZX80.
Ch.2 - Theory of Computers
Ch.3 - ZX80 BASIC.
Ch.4 - The ZX80 Monitor.
Ch.5 - Construction and Hardware
Ch.6 - ZX80 Programs.
App.1 - Comparison of Z-80 Opcodes and ZX80 Characters.
App.2 - The 8K ROM.

For a cassette of ten programs from the Companion add £4.95.

Cassette Software

All LINSAC program packs come on single C12 cassettes with printed run instructions. All apart from Games Pack 3 are designed for 1K ZX80's. Price £10.00 per pack including UK postage.

Games Pack 1 - Three Towers, Number Guessing, Mastermind, Sketcher, Hurkle, Nim, Symbol Simon.

Games Pack 2 - Nine Lives, The Maze Game, Plain Sailing, Noughts and Crosses, Chinese Puzzle, Tower of Hanoi, Battleships.

Games Pack 3 (for 2 - 4K ZX80's) - Fruit Machine, Four-in-a-Line, Zombies.

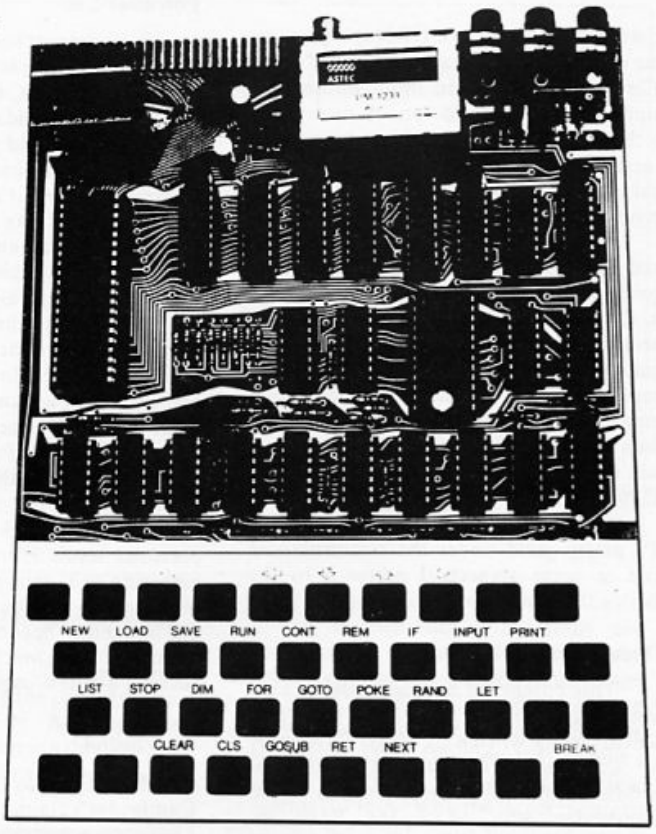
Education Pack 1 - Maths Drill, Dot Recognition, Musical Notes, Spelling Quiz, Day Finder.

Education Pack 2 - Graph Plotter, Prime Factors, Number Bases, Bar Charts, Statistics.

Utility Pack 1 - Memory Display, Hex Code Monitor, Renumber, Memory Search.

The above ZX80 products are distributed by IMAGE COMPUTING in the United States

The Times They are a' Changing: The Building of a MicroAce



Laura L. McLaughlin

Back in the beginnings of the personal computer industry, I built my first "very own" computer with a friend. That was in January of 1976 (less than 5 years ago). It was one of the first IMSAI computers (remember them?— they're one of many companies that are no longer). The box was about 19 1/2" X 17" X 7", weighed in excess of 40 pounds and contained a fromt panel, fan, power supply and a motherboard which would hold up to 22 S-100 boards. Initially we built 7 boards; a processor, four 4K RAM cards, a cassette interface and a serial interface. This was a many man-hours project— if my memory serves me correctly, I'd say about 80 man-hours— no small task. And at that point our only I/O on the system consisted of 8 switches on the front panel for entering one byte (character) at a time and 8 lights for output of byte. Of course we had a serial interface to drive a printer or terminal (maybe even with a keyboard), but that would have to wait until we could afford it. We could save a program, once loaded in through the switches, onto a cassette and then load it into memory from the cassette - but to do that we had to use the switches to load the program that could read the cassette. Please note that at this point we had invested approximately $2000.00 in our system.

Now consider this. Today I sat down to build another personal computer, a MicroAce that retails for $169.00 (less than 9% of the cost of the IMSAI), and takes approximately 7 hours (less than 10% of the time) to build. It measures about 8-1/2" X 7" X 1-1/2" and weighs in at 13 ounces.

But what do I get with the MicroAce? Well, let's look at that for a moment. It comes with processor, cassette interface, pressure-sensitive keyboard, video interface (with built in UHF modulator so that it connects directly to the antenna leads on a standard TV set), power supply (an AC adaptor just like the one you use on your calculator that plugs into a regular 110 volt wall outlet), 2K of RAM memory and a 4K Basic in ROM. Basic? When we first built the IMSAI we couldn't even buy a Basic that would run on it! The IMSAI had more memory and certainly more potential to expand (and, in fact, was eventually developed into an extremely powerful system), but in comparing what is available today versus what there was back in the "early" days of home computers, there is no question that the MicroAce is amazing.

I should mention that my background is in software; I am not a hardware person. Extensive exposure to the equipment has given me a reasonable understanding of how it works, but I am not an expert. The only experience I have had with building electronic kits is that with the IMSAI.

Okay, now that we've set the scene, let's get down to the discussion of what it took to build this machine. The first phase of the project was to read through the manual. Upon opening the booklet I first saw a xeroxed sheet with a heading that said "Important— Read this and the last page in the manual first."

Alright, this is a newly released product so I guess errata sheets are to be expected. The best way to handle an errata sheet is to transfer the corrections directly into the manual with a colored pen. Don't be afraid to write in the book— remember, it is yours. This way you will not forget (or lose) the changes. One of the changes involved installing two jumper wires on the back of the printed curcuit card (PCB), but nowhere did it say when this should be done. Since they would likely be in the way during normal assembly, I noted in the instructions (on page 11) that this should be done just prior to inserting the integrated circuits (ICs).

Now to the assembly instructions themselves. Note that the manual is sub-titled "The Teach-Yourself BASIC Manual." This is not particularly misleading. Of the 66 pages, the chapter on assembly takes 10; and that contains 4 pages on tips and component identification, 2 on board diagrams and 1 on a parts list. Much of the remaining 3 pages are either diagrams or precautions. The actual assembly instructions basically just say:

1. Install sockets
2. Install small discreet components
3. Install tape and power sockets, voltage regulator and modulator
4. Make tape and video cables (if necessary)
5. Insert ICs

These are indeed sparse assembly instructions. Now, I fully admit that the rest of the manual goes on to explain Basic to the novice in a very reasonable fashion, but it seems to me that a little more detail on the actual construction is called for (maybe even just a checklist by component type?). In any case, they do provide you with some very good information on both component identification and on many of the precautions that should be taken in the process of putting together the MicroAce (or any other digital electronic circuit board kit), including such things as how to avoid destroying static sensitive chips. I do hope you have some familiarity with the terminology, otherwise you may find the instructions confusing.

Having become familiar with the instrictions, I proceeded to compare the parts supplied against those in the component list. This was a relatively straightforward task since they provide an excellent section on parts identification, including a complete list of resistors, by number, with their color codes. Checking the parts off as I went along, and noting any discrepancies, I found I had 1 extra resistor, 1 extra socket and 2 missing diodes. Well the extras would obviously not be a problem, but the elusive diodes might.

Since the next step was to locate the positions of all the parts on the board (using the component layout on page 7 of the manual), I decided to look first for where the diodes were located. Low and behold, I saw markings for only 9 of them, instead of the 11 indicated in the parts list. Encouraging. Maybe I wasn't missing anything afterall. As a double check, I went to the schematic (pages 34-35) and once again could see only 9 diodes. It seemed safe to assume that the component list was wrong.

I found the rest of the parts easily. There was just one confusing marking on the board layout. Since this is a new release, it did not surprise me to find extra screenings on the board for parts no longer necessary. These were all clearly noted in the list of components. However, I noticed a symbol for a transistor (TR1) on the upper righthand corner. Once again I went to the schematic, but could find no indication of any transistors. With none supplied and no other mention of it to be found, I assumed it to be one more unused part, so I indicated that on my component list.

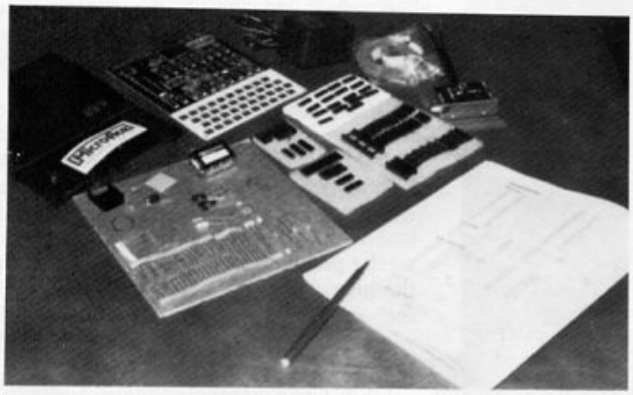


Photo 1 Checking for all of the parts

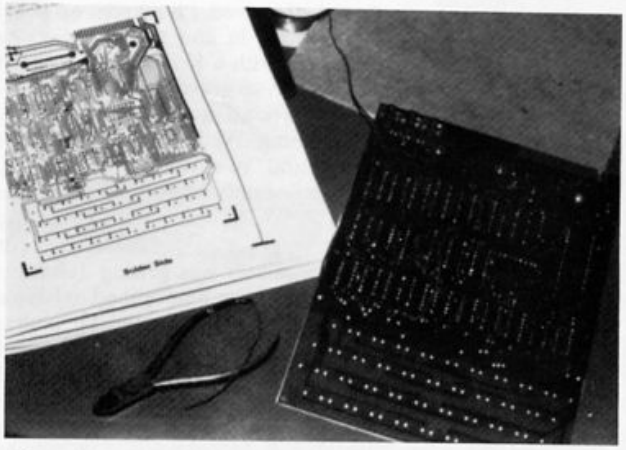
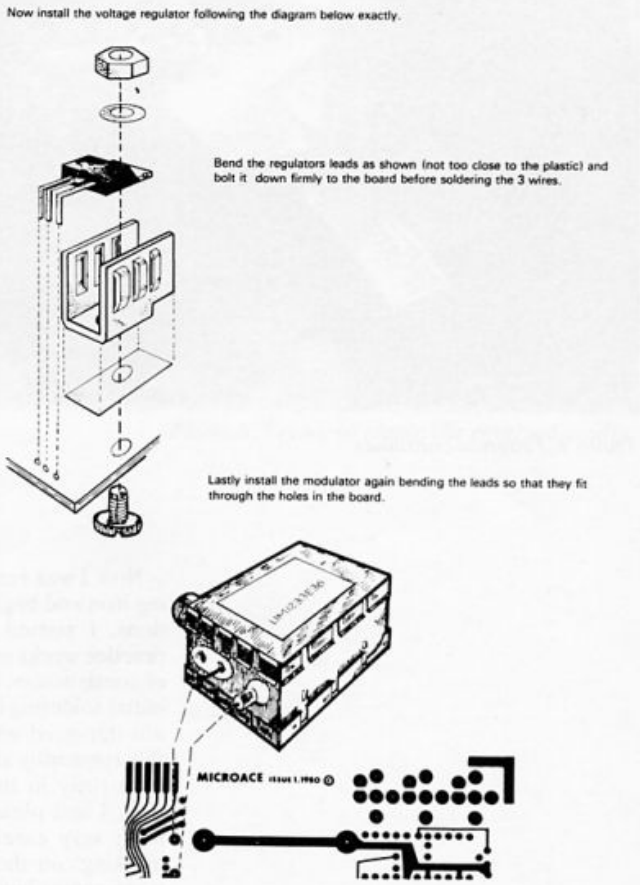


Photo 2 The solder side of the MicroAce



Sample page from assembly manual.

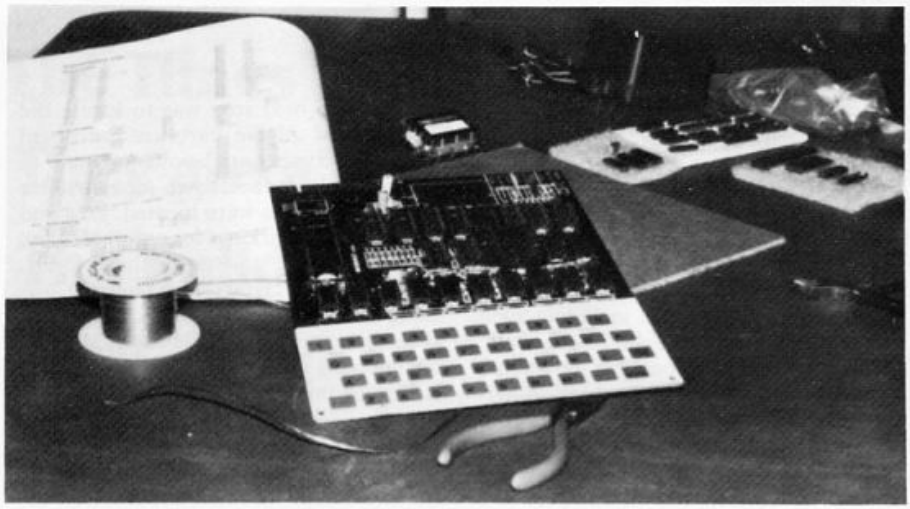


Photo 3 Beginning the assembly

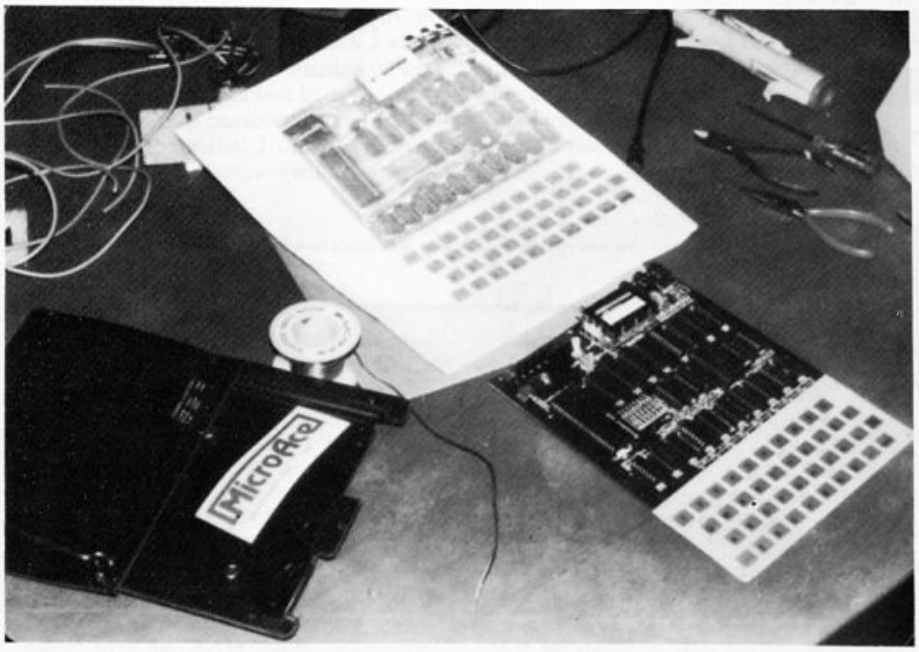


Photo 4 Progress continues

Now I was ready to turn on my soldering iron and begin. Following the instructions, I started with the sockets. This practice works well not only for efficiency of construction, but also because it means initial soldering is done on things that are not damaged when overheated. I found this reassuring since I was afraid I'd be a little rusty in this department. All went well. I was pleased to see that they had been very careful in the placement of markings on the board. Although things were somewhat tight in places, nothing was hidden as I put in the sockets.

The next task, according to the manual, was to install the discreet components. With no detailed instructions or checklist, I decided to follow the component list, checking off the parts as I put them on the board. Once again the component identification write-up turned out to be very useful, as it showed orientation markings for those parts that had to be inserted in a particular direction (the two electrolytic capacitors and the nine diodes). First I put in all the resistors (R1-R36), double checked their positioning, soldered them in place, clipped their leads and inspected the solder joints. I then did the same with the capacitors (C1-C16), then the diodes (D1-D9) and the crystal (X1).

The instructions then call for installing the 3 tape and power sockets (not mentioned in the component list), the voltage regulator (U25) and the UHF modulator, in that order. Following the excellent diagrams in the manual, this was done with no problem. Then came the tape and video cables. The manual states that if you're lucky (based on availability) they will come pre-assembled. I was not lucky. But since I had cables available from other equipment, I postponed this task for later. I do feel that if they are going to send unassembled cables, they should at least provide a diagram for their construction.

In any case, all I had left to do was the insertion of the ICs. Or did I? No, I had noted in my manual that this was the best time to install the two jumpers called for by the errata sheet. After doing that, I inserted the 24 ICs, being careful to follow all the precautions and tips mentioned in the instructions (except for earthing myself on a cold water tap; I discovered 5 years ago that working barefoot on a tiled floor eliminated the risk of static).

I now had a completed MicroAce computer. It seemed sensible to test it before putting it into its case, so the crucial moment had arrived. I hooked it up to my TV, plugged it in and there it was— a blank screen with a K in the bottom left corner. So far, so good.

Then I pressed a key. Nothing happened. Nothing. Well, it was time to start checking and re-checking the board. After spending quite some time doing just that, looking for misplaced or incorrectly oriented components, bad solder joints or bent under pins, I was getting quite discouraged. Yet I was determined to have a working computer. Sure, the kit comes with a terrific guarantee— for a flat fee of $20 you can return it to the company and they will repair it and send it back in perfect working order. But that might take a couple of weeks. I wanted to start using it immediately.

As I said before, I am not a hardware person, but I decided to look at the schematic anyway to see if I could pin-point, or at least isolate, the problem. I noticed immediately that the 8 diodes along the bottom of the board were a direct interface between the keyboard and the rest of the circuitry (going specifically to U11). I had checked those diodes a number of times already and was sure they were installed correctly. Checking them again reassured me that I was right.

I won't go through the whole frustrating story of the next few hours trying to track down the problem with a friend (who is a hardware person and who also checked the diodes). Needless to say, in the end we discovered that the diodes were in backwards. What had appeared to be a yellow diode with a green band on one end, was in reality a clear diode with a narrow green band on one end and a very wide yellow one on the other. What can I say but that I hope someone can benefit from my mistake.

Anyway, at that point (about midnight), it didn't matter. My MicroAce worked—that was exciting. Forgetting all the earlier frustration, I put it in its case and sat down to a couple of hours of joyous programming.

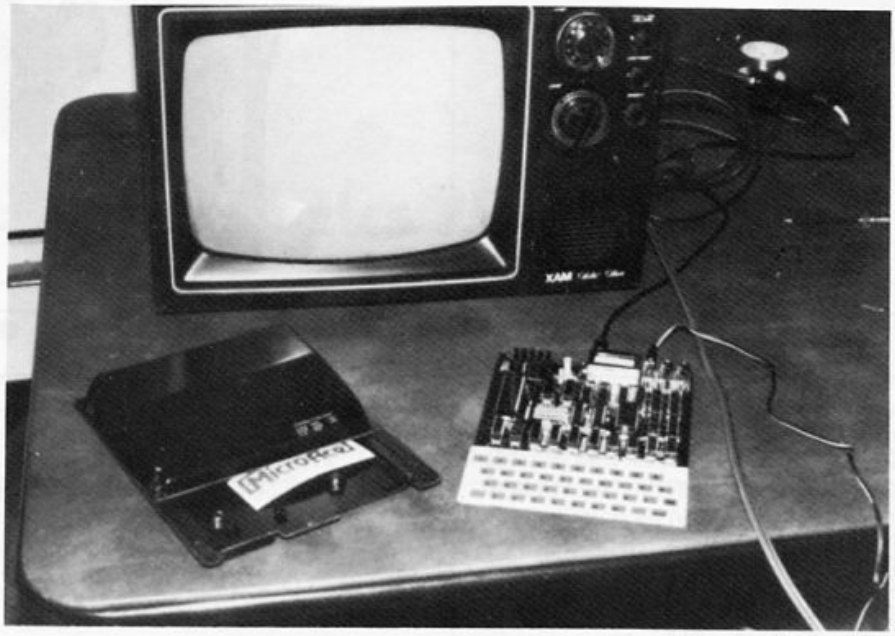


Photo 5 Almost done

Photo 6 Trying to locate the reversed diodes

Photo 7 The finished product

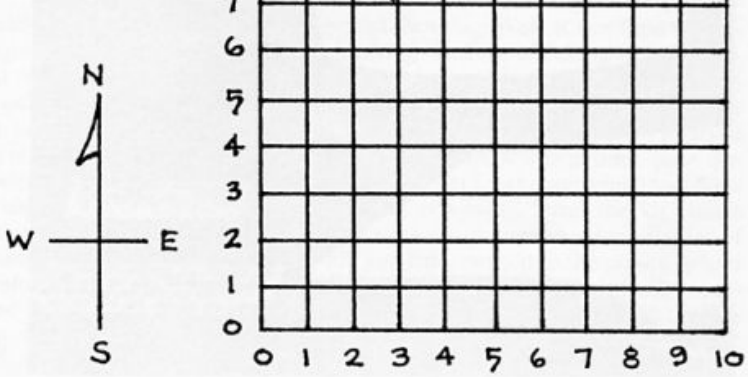
Hurkle

```
220 PRINT "A HURKLE IS HIDING IN A 10 BY 10 GRID."
240 PRINT "TRY TO GUESS THE LOCATION OF THE HURKLE."
250 PRINT "YOU GET 5 TRIES."
260 PRINT "AFTER EACH TRY, I WILL TELL YOU"
270 PRINT "THE DIRECTION TO GO TO FIND HIM."
280 PRINT "HIT NEWLINE TO START."
282 INPUT A$
283 CLS
284 RANDOMISE
285 LET A=RND(10)
286 LET B=RND(10)
310 FOR K=1 TO 5
320 PRINT "GUESS NUMBER " ;K
325 PRINT "X COORDINATE?"
330 INPUT X
333 PRINT "Y COORDINATE?"
335 INPUT Y
340 IF ABS (X-A)+ABS(Y-B)=0 THEN GO TO 500
350 CLS
355 PRINT "X=";X;"Y=";Y;"  ";
360 GO SUB 610
380 NEXT K
420 PRINT "SORRY, THAT IS 5 GUESSES"
430 PRINT "THE HURKLE IS HIDING AT ";A;",";B
440 PRINT
450 PRINT "PLAY AGAIN?"
460 INPUT A$
470 IF NOT CODE(A$)=62 THEN STOP
480 GO TO 283
500 PRINT
520 PRINT "YOU FOUND HIM IN ";K;" GUESSES."
540 GO TO 440
610 PRINT "GO ";
620 IF Y=B THEN GO TO 670
630 IF Y<B THEN GO TO 660
640 PRINT "SOUTH";
650 GO TO 670
660 PRINT "NORTH";
670 IF X=A THEN GO TO 720
680 IF X<A THEN GO TO 710
690 PRINT "WEST"
700 GO TO 720
710 PRINT "EAST"
720 PRINT
730 RETURN
```

A Hurkle is a happy beast and lives in another galaxy on a planet named Lirht that has three moons. Hurkle are favorite pets of the Gwik, the dominant race of Lirht and . . .well, to find out more, read "The Hurkle is a Happy Beast," in the book *A Way Home*, by Theodore Sturgeon.

In this program a shy Hurkle is hiding on a 10 by 10 grid. Homebase is point 0,0 in the southwest corner. Your guess as to the gridpoint where the Hurkle is hiding should be a pair of whole numbers. After each try the computer will tell you the approximate direction to go look for the Hurkle. You get five guesses to find him.

This program was written by Bob Albrecht of People's Computer Company, and was translated by Laura McLaughlin.



Converting From Other Basics

David Lubar

While Basic is a distinct computer language, it exists in many forms. Thus, a program written for another machine might have to be converted before it will work with the Sinclair. This series will deal with conversion techniques, in hopes of allowing you to get as many programs as possible running.

A command found in many Basics is STEP. This is used in FOR-NEXT loops when an increment other than one is desired. For example:

```
10 FOR = 1 TO 9 STEP 2
20 PRINT I
30 NEXT I
```

This program will print the odd numbers from one to nine. At each pass through the loop, the variable I will increased by two. STEP can also be negative:

```
10 FOR I = 10 TO 1 STEP -1
20 PRINT I
30 NEXT I
```

In this case, I will decrease by one on each pass through the loop.

There is no STEP command in Sinclair Basic. Fortunately, there is an easy way around this. Loops are convenient, but they can always be replaced with IF-THEN statements. This is done by using a variable for a counter, and checking to see whether it is still in the desired range. If it is, the program portion is repeated. A simple loop such as

```
10 FOR I = 1 TO 10
20 PRINT I
30 NEXT I
```

could be replaced with

```
10 LET I =1
20 PRINT I
30 LET I = I + 1
40 IF I < 11 THEN GOTO 20
```

Line 10 initializes the variable. After the desired action is performed by line 20, line 30 increases the variable. Then, at line 40, the variable is checked to determine whether it is still within the set limits. A value of 11 was used since we want the action to be performed ten times. The variable is increased and checked after the action since this is the way it is done when using FOR-NEXT loops. As might be obvious now, the STEP command is equivalent to the actions performed by line 30. In this case, the STEP was one. For other steps, all that is needed is a change of the values used. For instance, the first example could be rewritten as:

```
10 LET I = 1
20 PRINT I
30 LET I = I + 2
40 IFI < 10 THEN GOTO 20
```

For the second example, with a negative STEP, you would use

```
10 LET I = I
20 PRINT I
30 LET I = I - 1
40 IF I > 0 THEN GO TO 20
```

Notice that when replacing the negative STEP, a > was used, while a < is used in positive STEPS. In either case, you have to check to see if the value has gone beyond the desired range.

Some programs use fractional values for STEP. In some cases you can get around this by multiplying everything by a constant that will make the STEP value an integer. In other cases, where the decimal value is essential, this won't work.

Future articles will cover other conversion techniques, showing how to get those strange looking programs to run on the ZX-80.

Editorial

by David Lubar

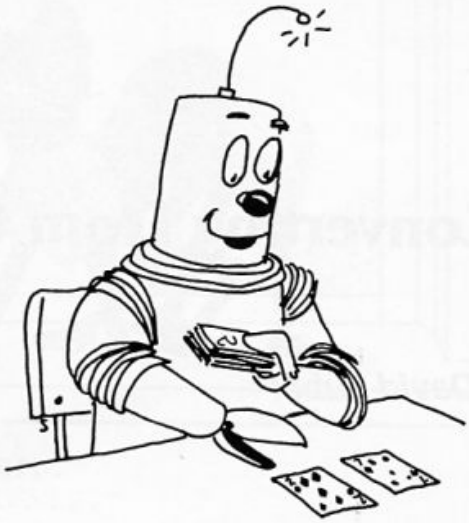
The Sinclair ZX80 is the new kid in town; a short-lived position in this rapidly changing field. In the past few years, several dozen personal computers have made an appearance. Some are thriving, others have fallen into the obscurity of bargain sales. We believe that the Sinclair is going to be one of the winners; otherwise, there would be no sense in starting this magazine. So, what are our plans? We aim to cover many levels, with articles that will help beginners, as well as features for pros.

The Sinclair is an exciting machine with a lot of potential, and we hope to reflect this in the programs and articles we publish. There will be plenty of games, both new ones and conversions of classics. As new aspects or hidden capabilities of the Sinclair are discovered, we'll pass the information on to you. As you make discoveries, we hope you'll take the time to write articles for us.

There is another question tied in with this: what happens when Sinclair owners begin to get extended Basic and expanded memory? Inevitably, there will be a time when our readers cover the full spectrum from those with minimal systems to those with fully expanded ones. This might seem to pose a problem, but the problem is not unique. Several of the popular home computers started out as very small systems with little memory and modest Basics. As memory became less expensive, most owners expanded their systems. For a time, computer magazines dealt mostly with the original configurations, and occasionally ran an article or program aimed at expaned systems. Later, the coverage shifted, but the smaller systems were not ignored.

This will probably be the case with SYNC. We'll stick to the 1K RAM 4K Basic machine now, while running a few pieces designed for other configurations. Later, there will most likely be more attention given to expanded systems, but plenty of coverage will still be dedicated to the original model. In a way, this will be determined by you. Much of our material will come from free-lance submissions. As more writers expand their Sinclairs, we will receive more articles and programs aimed at larger systems. But it must be stressed that SYNC will never ignore those pioneers who own the original model.

Dealing Out a Conversion



David Lubar

Sinclair owners will probably spend a fair amount of time converting programs from other Basics. This can be a rewarding and educational practice or an exercise in frustration, depending on how you go about it. To pass on a few tips and tricks for translating, I'll explain the techniques used in getting *Acey Deucy* running on the ZX 80.

The first step in any translation is to get a good idea of the functions being performed by the program, and of the logic flow behind these functions. Let's go through the original listing, examine what each part does, and see how the functions can be duplicated on the Sinclair. Basically, the program does the following; it deals two cards, accepts a bet, deals a third card, checks for a win or loss, adjusts the player's total, and checks to see if he is broke. The first portion of the program prints the instructions. In the interest of saving memory, these lines were not included in the translation. If there is room, they can always be added later.

In lines 100 and 110, variables are defined. "N" is not used later in the program, so it isn't included in the translation. The original programmer probably had plans to use it in some later version of the game and didn't bother to remove it from the listing. "Q" holds the amount of money the player has. The next section, lines 120 and 130, print the player's total and then print a blank line. While the extra PRINT statement adds readability, it also uses memory. Like the instructions, it can always be added later. The program then jumps to the section from 260 to 650 where the cards are selected and displayed. This is where the real changes begin to occur.

```
10 PRINT TAB(26);"ACEY DUCEY CARD GAME"
20 PRINT TAB(15);"CREATIVE COMPUTING  MORRISTOWN, NEW JERSEY"
21 PRINT
22 PRINT
23 PRINT
30 PRINT"ACEY-DUCEY IS PLAYED IN THE FOLLOWING MANNER "
40 PRINT"THE DEALER (COMPUTER) DEALS TWO CARDS FACE UP"
50 PRINT"YOU HAVE AN OPTION TO BET OR NOT BET DEPENDING"
60 PRINT"ON WHETHER OR NOR YOU FEEL THE CARD WILL HAVE"
70 PRINT"A VALUE BETWEEN THE FIRST TWO."
80 PRINT"IF YOU DO NOT WANT TO BET, INPUT A 0"
100 N=100
110 Q=100
120 PRINT"YOU NOW HAVE ";Q;" DOLLARS"
130 PRINT
140 GOTO 260
210 Q=Q+M
220 GOTO 120
240 Q=Q-M
250 GOTO 120
260 PRINT"HERE ARE YOUR NEXT TWO CARDS "
270 A=INT(14*RND(1))+2
280 IF A<2 THEN 270
290 IF A>14 THEN 270
300 B=INT(14*RND(1))+2
310 IF B<2 THEN 300
320 IF B>14 THEN 300
330 IF A>=B THEN 270
350 IF A<11 THEN 400
360 IF A=11 THEN 420
370 IF A=12 THEN 440
380 IF A=13 THEN 460
390 IF A=14 THEN 480
400 PRINT A
410 GOTO 500
420 PRINT"JACK"
430 GOTO 500
440 PRINT"QUEEN"
450 GOTO 500
460 PRINT"KING"
470 GOTO 500
480 PRINT"ACE"
500 IF B<11 THEN 550
510 IF B=11 THEN 570
520 IF B=12 THEN 590
530 IF B=13 THEN 610
540 IF B=14 THEN 630
550 PRINT B
560 GOTO 650
570 PRINT"JACK"
580 GOTO 650
590 PRINT"QUEEN"
600 GOTO 650
610 PRINT"KING"
620 GOTO 650
630 PRINT"ACE"
640 PRINT
650 PRINT
660 INPUT"WHAT IS YOUR BET";M
670 IF M<>0 THEN 680
675 PRINT"CHICKEN!!"
676 PRINT
677 GOTO 260
680 IF M<=Q THEN 730
690 PRINT"SORRY, MY FRIEND BUT YOU BET TOO MUCH"
700 PRINT"YOU HAVE ONLY ";Q;" DOLLARS TO BET"
710 GOTO 650
730 C=INT(14*RND(1))+2
740 IF C<2 THEN 730
750 IF C>14 THEN 730
760 IF C<11 THEN 810
770 IF C=11 THEN 830
780 IF C=12 THEN 850
790 IF C=13 THEN 870
800 IF C=14 THEN 890
810 PRINT C
820 GOTO 910
830 PRINT"JACK"
840 GOTO 910
850 PRINT"QUEEN"
860 GOTO 910
870 PRINT"KING"
880 GOTO 910
890 PRINT "ACE"
900 PRINT
910 IF C>A THEN 930
920 GOTO 970
930 IF C>=B THEN 970
950 PRINT"YOU WIN!!!"
960 GOTO 210
970 PRINT"SORRY, YOU LOSE"
980 IF M<Q THEN 240
990 PRINT
1000 PRINT
1010 PRINT"SORRY, FRIEND BUT YOU BLEW YOUR WAD"
1020 INPUT"TRY AGAIN (YES OR NO)";A$
1030 IF A$="YES" THEN 110
1040 PRINT"OK HOPE YOU HAD FUN"
1050 END
```

Listing of Acey Ducey from *Basic Computer Games*. Copyright 1976 by Creative Computing. Reprinted with permission.

```
          ACEY DUCEY CARD GAME
CREATIVE COMPUTING  MORRISTOWN, NEW JERSEY


ACEY-DUCEY IS PLAYED IN THE FOLLOWING MANNER
THE DEALER (COMPUTER) DEALS TWO CARDS FACE UP
YOU HAVE AN OPTION TO BET OR NOT BET DEPENDING
ON WHETHER OR NOR YOU FEEL THE CARD WILL HAVE
A VALUE BETWEEN THE FIRST TWO.
IF YOU DO NOT WANT TO BET, INPUT A 0
YOU NOW HAVE  100  DOLLARS

HERE ARE YOUR NEXT TWO CARDS
 2
 9

WHAT IS YOUR BET? 25
QUEEN
SORRY, YOU LOSE
YOU NOW HAVE  75  DOLLARS


HERE ARE YOUR NEXT TWO CARDS
 4
 10

WHAT IS YOUR BET? 25
 10
SORRY, YOU LOSE
YOU NOW HAVE  50  DOLLARS

HERE ARE YOUR NEXT TWO CARDS
 8
QUEEN

WHAT IS YOUR BET? 0
CHICKEN!!

HERE ARE YOUR NEXT TWO CARDS
 6
 10

WHAT IS YOUR BET? 20
 10
SORRY, YOU LOSE
YOU NOW HAVE  30  DOLLARS
```

sample run from original Acey Ducey game in *Basic Computer Games*.

Several things are happening here. First, a value from two to fourteen is chosen. By using fourteen as the value of an ace (aces are high), the cards can be compared easily. In the original program, the first card (variable A) had to be lower in value than the second card (variable B). If this wasn't the case, new values had to be used. In the conversion, it doesn't matter which of the two is higher. The program only checks to see if they are the same. In this case, since the player can't win, new values are selected (see line 300 of the translation). Next, the values are printed. If the value is below eleven, a number is used. For a higher value, the name of the card is needed. In the original, this is done separately for each card. To save space in the conversion, this function was put into a subroutine which could be used for all three cards. The subroutine uses the variable "C". In line 320 of the conversion, C is made equal to A. Thus the subroutine prints the name of the card if A is greater than ten.

At line 660 of the original, the bet is taken. Line 670 shows a common operation used in many programs: an inequality where more than one operator is used. The symbols =, < , and > are often used in pairs. In this case, the symbols "< >" mean "not equal to." This can be translated using the NOT operator. Whenever a program has two symbols (such as >=, which means "greater than or equal to,") the line can be translated using NOT and the unused symbol of the trio <, >, and =. So a line such as

10 IF X >= 10 THEN GO TO 50

can be translated as

10 IF NOT X < 10 THEN GO TO 50

Getting back to the program, line 680 checks to see if the player bet more than he has in his winnings. This line could have been translated using the trick just mentioned. For variety, another method was used. In this instance, the two symbols were used, separated by a logical OR (see line 410 of the translation). Lines 730 through 890 select and print the third card. In the translation, this is replaced with 450 through 465. Note that the lines in the original, such as 740, that check for values less than two or greater than fourteen aren't needed. The RND function won't return a value outside the desired range.

After the third card is displayed, the values must be compared to see whether there is a win or a loss. In the original, since A was always less than B, the program only had to determine whether D was greater than A and less than B. This is done in the series of lines from 910 to 970. In the translation, we have to see whether D is between A and B, regardless of whether A or B is the higher of the original pair. This is done in line 600. D must either be greater than A and less than B or greater than B and less than A. Otherwise, the player loses.

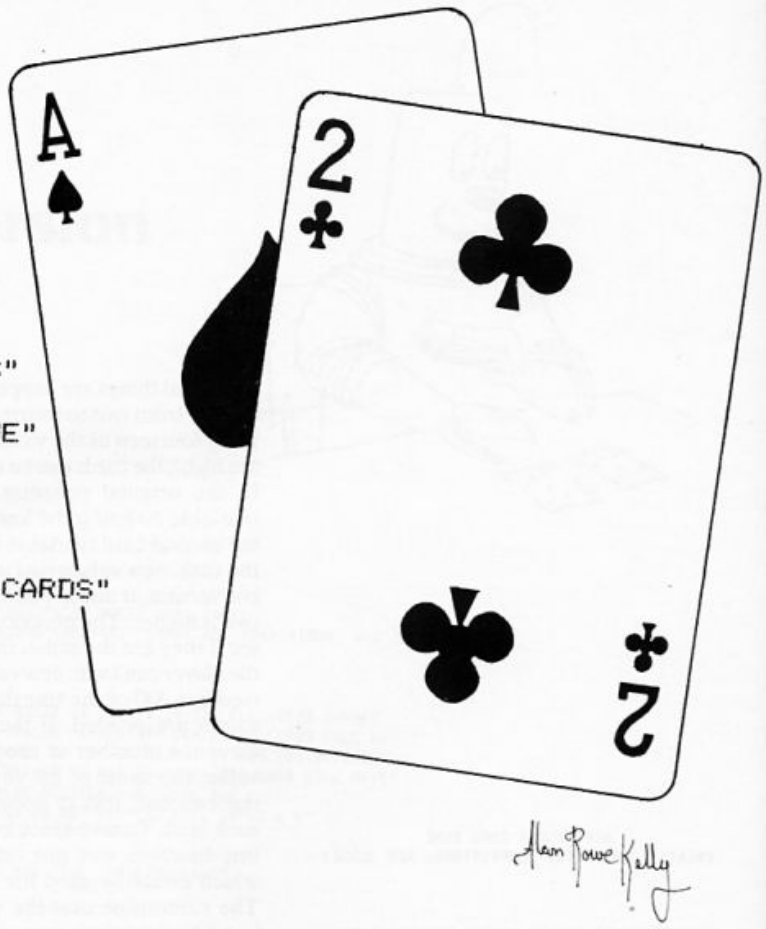
Once a win or loss has been determined, the program must adjust the player's total. In the original, different routines were used for win and loss. This is the section from 210 to 240. To save space, just one routine was used in the translation. This is done in line 215. If the player lost, M will have a negative value (see line 630) and the bet will actually be subtracted from the total.

Line 620 of the translation checks to see if the player is broke. If he is, the program goes on to line 900 where the player is offered another chance.

That about covers the translation. Note that on each pass through the game, the player must hit NEWLINE to continue (line 210). This serves two purposes. The input stops the program temporarily, allowing the printout to appear, and it also gives the player a chance to read everything before the CLS is encountered. Also, commas are used in the print routines so the cards will be printed on one line.

Those who are interested might try adding some improvements to the game. For example, in some versions of Acey Deucy, the player loses $5 whenever a pair is dealt. Also, in the real version, the player has a choice of letting an ace be either high or low if it is the first card of the two dealt on any hand. These features could probably be added to the existing program with little difficulty.

Acey Ducey




```
110 LET Q=100
115 RANDOMISE
120 PRINT "YOU HAVE ";Q;" DOLLARS"
140 GO TO 260
210 PRINT "HIT NEWLINE TO CONTINUE"
211 INPUT A$
214 CLS
215 LET Q=Q+M
220 GO TO 120
260 PRINT "HERE ARE THE NEXT TWO CARDS"
270 LET A=RND(13)+1
280 LET B=RND(13)+1
300 IF A=B THEN GO TO 270
310 IF A<11 THEN PRINT A,
320 LET C=A
330 GO SUB 500
340 IF B<11 THEN PRINT B;
350 LET C=B
360 GO SUB 500
365 PRINT
370 PRINT "YOUR BET?"
380 INPUT M
390 IF NOT M=0 THEN GO TO 410
395 CLS
400 PRINT "CAUTIOUS"
405 GO TO 250
410 IF M=Q OR M<Q THEN GO TO 450
420 PRINT "SORRY, YOU ONLY HAVE ";Q;" DOLLARS TO BET."
430 GO TO 370
450 LET D=RND(13)+1
460 IF D<11 THEN PRINT D
465 LET C=D
470 GO SUB 500
480 GO TO 600
500 IF C=11 THEN PRINT "JACK",
510 IF C=12 THEN PRINT "QUEEN",
520 IF C=13 THEN PRINT "KING",
530 IF C=14 THEN PRINT "ACE",
540 RETURN
600 IF D>A AND D<B OR D<A AND D>B THEN GO TO 700
610 PRINT "SORRY, YOU LOSE"
620 IF M>Q OR M=Q THEN GO TO 900
630 LET M=-M
640 GO TO 210
700 PRINT "YOU WIN"
710 GO TO 210
900 PRINT "YOU ARE TAPPED OUT. PLAY AGAIN?"
910 INPUT A$
920 IF NOT CODE (A$)=62 THEN STOP
990 GO TO 110
```

This simulation of the Acey Ducey card game. In the game, the dealer (the computer), deals two cards face up. You have an option to bet or not to bet depending whether or not you feel the next card dealt will have a value between the first two.

Your initial money is set at $100. The game keeps going until you lose all your money, or interrupt the program.

The original program author was Bill Palmby of Prairie View, Illinois, and David Lubar translated for the ZX-80.

The first personal computer for under $200.

The Sinclair ZX80. A complete computer—only $199.95 *plus $5.00 shipping.*

Now, for just $199.95, you can get a complete, powerful, full-function computer, matching or surpassing other personal computers costing several times more.

It's the Sinclair ZX80. The computer that "Personal Computer World" gave 5 stars for 'excellent value.'

The ZX80 cuts away computer jargon and mystique. It takes you straight into BASIC, the most common, easy-to-use computer language.

You simply take it out of the box, connect it to your TV, and turn it on. And if you want, you can use an ordinary cassette recorder to store programs. With the manual in your hand, you'll be running programs in an hour. Within a week, you'll be writing complex programs with confidence.

All for under $200.

Sophisticated design makes the ZX80 easy to learn, easy to use.

We've packed the conventional computer onto fewer, more powerful LSI chips—including the Z80A microprocessor, the faster version of the famous Z80. This makes the ZX80 the world's first truly portable computer (6½" x 8½" x 1½" and a mere 12 oz.). The ZX80 also features a touch sensitive, wipe-clean keyboard and a 32-character by 24-line display.

Yet, with all this power, the ZX80 is easy to use, even for beginners.

Your course in computing.

The ZX80 comes complete with its own 128-page guide to computing. The manual is perfect for both novice and expert. For every chapter of theory, there's a chapter of practice. So you learn by doing—not just by reading. It makes learning easy, exciting and enjoyable.

You'll also receive a catalog packed with items that can make your ZX80 even more useful. Including 27 program cassettes, from games and home budgeting for just $6.95, to Sinclair's unique Computer Learning Lab. And books, hardware options and other accessories.

ZX80's advanced design features.

Sinclair's 4K integer BASIC has performance features you'd expect only on much larger and more expensive computers.

- Unique 'one touch' entry. Key words (RUN, PRINT, LIST, etc.) have their own single-key entry to reduce typing and save memory space.
- Automatic error detection. A cursor identifies errors immediately to prevent entering programs with faults.
- Powerful text editing facilities.
- Also programmable in machine code.
- Excellent string handling capability—up to 26 string variables of any length.
- Graphics, with 22 standard symbols.
- Built-in random number generator for games and simulations.

Sinclair's BASIC places no arbitrary restrictions on you—with many other flexible features, such as variable names of any length.

And the computer that can do so much for you now will do even more in the future. Options will include expansion of 1K user memory to 16K, a plug-in 8K floating-point BASIC chip, applications software, and other peripherals.

Order your ZX80 now!

The ZX80 is available only by mail from Sinclair, a leading manufacturer of consumer electronics worldwide.

To order by mail, use the coupon below. But for fastest delivery, order by phone and charge to your Master Charge or VISA. The ZX80 is backed by a 30-day money-back guarantee, a 90-day limited warranty with a national service-by-mail facility, and extended service contracts are available for a minimal charge.

Price includes TV and cassette connectors, AC adaptor, and 128-page manual.

All you need to use your ZX80 is a standard TV (color or black and white). The ZX80 comes complete with connectors that easily hook up to the antenna terminals of your TV. Also included is a connector for a portable cassette recorder, if you choose to store programs. (You use an ordinary blank cassette.)

The ZX80 is a family learning aid. Children 10 and above will quickly understand the principles of computing—and have fun learning.

Master Charge or VISA orders call: (203) 265-9171. We'll refund the cost of your call.

Information: General and technical—(617) 367-1988, 367-1909, 367-1898, 367-2555. Phones open Monday-Friday from 8 AM to 8 PM EST.

sinclair

Sinclair Research Ltd., 475 Main St., P.O. Box 3027, Wallingford, CT 06492.

To: **Sinclair Research Ltd., 475 Main St., P.O. Box 3027, Wallingford, CT 06492.**

Please send me _____ ZX80 personal computer(s) at $199.95* each (US dollars), plus $5 shipping. (Your ZX80 may be tax deductible.)

I enclose a check/money order payable to Sinclair Research Ltd. for $_____.

Name _____

Address _____

City _____ State _____ Zip _____

Occupation: _____ Age: _____

Intended use of ZX80: _____

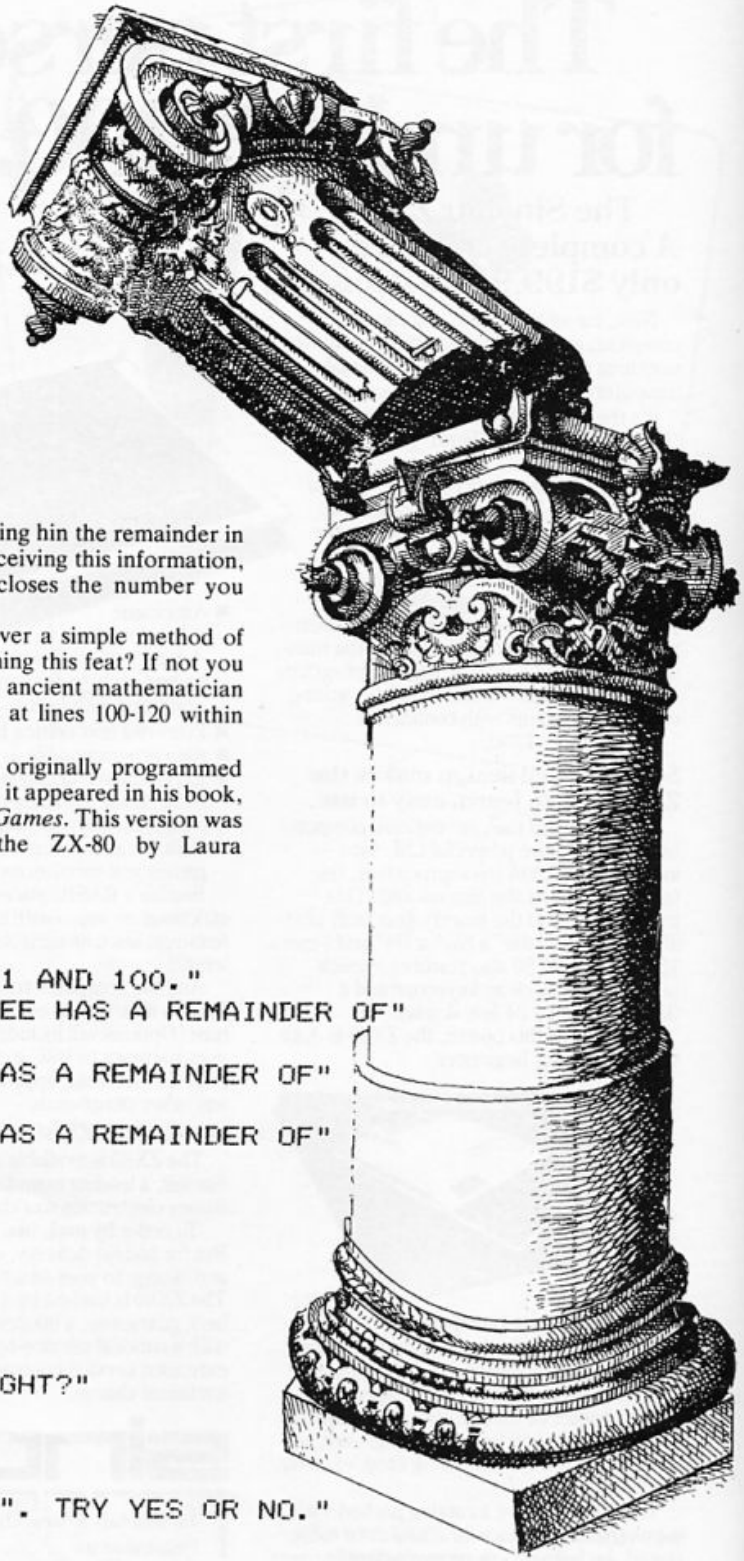
Have you ever used a computer? ☐ Yes ☐ No.

Do you own another personal computer? ☐ Yes ☐ No.

*For Conn. deliveries, add sales tax.

SY-1-1

Nicomacus



One of the most ancient forms of arithmetical puzzle is sometimes referred to as a "boomerang." At some time, everyone has been asked to "think of a number," and, after going through some process of private calculation, to state the result, after which the questioner promptly tells you the number you originally thought of. There are hundreds of varieties of the puzzle.

The oldest recorded example appears to be that given in *Arithmetica* of Nicomachus, who died about the year 120. He tells you to think of any whole number between 1 and 100 and divide it successively by 3, 5, and 7, teling hin the remainder in each case. On receiving this information, he promptly discloses the number you thought of.

Can you discover a simple method of mentally performing this feat? If not you can see how the ancient mathematician did it by looking at lines 100-120 within the program.

The game was originally programmed by David Ahl and it appeared in his book, *Basic Computer Games*. This version was translated for the ZX-80 by Laura McLaughlin.

```
30 PRINT "THINK OF A NUMBER BETWEEN 1 AND 100."
40 PRINT "YOUR NUMBER DIVIDED BY THREE HAS A REMAINDER OF"
45 INPUT A
50 PRINT "YOUR NUMBER DIVIDED BY 5 HAS A REMAINDER OF"
55 INPUT B
60 PRINT "YOUR NUMBER DIVIDED BY 7 HAS A REMAINDER OF"
65 INPUT C
70 PRINT
80 PRINT "LET ME THINK A MOMENT..."
100 LET D=70*A+21*B+15*C
110 IF NOT D>105 THEN GO TO 140
120 LET D=D-105
130 GO TO 110
140 PRINT "YOUR NUMBER WAS ";D;", RIGHT?"
160 INPUT A$
170 IF CODE (A$)=62 THEN GO TO 220
180 IF CODE(A$)=51 THEN GO TO 240
190 PRINT "I DO NOT UNDERSTAND ";A$;". TRY YES OR NO."
200 GO TO 160
220 PRINT "HOW ABOUT THAT."
230 GO TO 250
240 PRINT "I THINK YOUR ARITHMETIC IS IN ERROR."
250 PRINT
260 PRINT "WANT TO TRY ANOTHER?"
270 INPUT A$
280 IF NOT CODE(A$)=62 THEN STOP
290 CLS
300 GO TO 30
```

A Weekend With the ZX-80

Joshua Singer

I have always been intimidated by computers, and mathematics in general. One look at the Sinclair ZX-80, and I was confused. It wasn't what I had expected; there were no blinking lights, or humming, and no rows of ceiling high, pulsing units. It looked nothing like my 1950's science fiction concept of a computer. The ZX-80 was a pleasant suprise. It didn't look intimidating. In fact, it looked like a machine I might even be able to operate. Another suprise: I lifted the computer and found that it weighed only a few ounces.

I borrowed the computer from a friend, took it home and immediately connected it to my battered, portable television, which showed the clearest picture ever to grace the screen. With the Sinclair, the T.V., and the owner's manual in front of me, I began my first lesson in programming. The manual tells you everything you need to know, in clear, concise, often amusing language. The manual was so interesting that I lost track of time, something I rarely do when dealing with numbers and other non-literary subjects.

The book is extremely logical, beginning with instructions on how to connect the Sinclair to your television set. The same section also explains using a cassette recorder to store programs. The introductory portion of the manual even includes an explanation of computer languages, ending with a discussion of the Basic used with the ZX-80. As a complete beginner, I related to the way in which the booklet explained Basic as a simplified English dialect. It made sense, maing me realize that to give the computer commands, I merely had to be concise, as well as careful in choosing my words. The book suggested the machine was stupid, and therefore, that one had to be careful about the manner in which it was addressed. I liked that attitude because it made me feel brighter than the computer.

The manual continued by explaining general programming procedures, and several computer terms. I was unable to work my way through the entire manual, but that was only because of my own time limitations. By the end of the first session I could program and run addition, multiplication, and division problems. Most of these were simple enough to do by hand, but were a good foundation for learning programming. The book eventally enables the user/reader to execute games and provides enough information for the Sinclair owner to continue the learning process individually.

With just two weekends of not terrribly intensive study, I can now run just about any pre-programmed cassette, as well as doing some of my own programming. I am no programmer, but I do find that I can have a lot of fun playing games or learning on the ZX-80. It is such an easy machine to learn, that I have no doubts that I'll be writing my own programs in the near future.

QUALITY SOFTWARE

Graphics;
($5) Poke to memory mapped screen - use your ZX80 as a sketching pad. Character and direction control.

Music ;
($15) Tunes to your cassette recorder, with selectable octaves, semitones and duration from a wide range of notes. Your tune will repeat itself continuously. For the best results, memory 1K is necessary.

Hexadecimal;
($5) Hex. Loader - this routine provides a simple means for machine code programming in Hexadecimal notation. A 'String Rotate' demonstration program is included. Hex. Display - this routine allows you to inspect the contents of memory in 64 byte blocks - just enter the decimal address.

Active Display;
($20) Write your own games or use the ZX80 for continuous data monitoring. You decide how long a display will hold before your program continues (1/60th second to approx. 16 mins.). A Hex. Loader and a 'top row' keyboard scanning routine is included with a demo.

Breakout;
($20) Try your skill at knocking all the bricks from the wall before your final ball is lost. The ball starts at '9' and decrements each time it is lost until the game is over. The paddle moves swiftly under your control - the game restarts automatically.

Movies ;
($15) Pictures of your own creation (7x8 characters) are displayed in rapid rotation giving animation effect. For the best results, memory 1K is necessary.

Space Intruders;
($25) A simplified version of the famous arcade game. Zap the intruders as they attempt to land. The impulse to beat your last score is paramount. Requires 2K memory.

No modifications to hardware whatsoever. All software is thoroughly tested and claimed to be the most advanced developed so far. Games are written in machine code to make maximum use of memory and provide continuous live action without loss of T.V. synchronisation. All programs run in 1K memory unless stated otherwise.

Inclusive prices for program listing, details and Air Mail postage, (Send only $90 total if all the software listed is ordered together). Cheques (or S.A.E. for U.K. price list and details) to:- K.Macdonald, 26 Spiers Close, Knowle, Solihull, West Midlands, B93 9ES, ENGLAND.

The SYNC Challenge

The editor wants you to prove your programming skills. We at SYNC have chosen Hammurabi from Creative Computing's *Basic Computer Games* as the challenge program because it is a complex and refined game. We challenge you to fit this complex game into the 1K memory of a ZX-80.

In the conversion we are seeking to preserve as much of the original game as possible. Therefore, the primary factor in our evaluation of your conversion will be the preservation of as many facets of the game as possible. Judging will also be based on the ease of operation and the demonstration of good programming technique.

All programs must be submitted on cassette, with a letter of explanation. Both the cassette and the letter must be clearly marked with the participant's name, address and phone number. Entries received by March 15, 1981 will be judged in time for publication in the May/June issue.

When Hammurabi is converted for the ZX-80, the program must be substantially shortened. Such things as the initial directions, copyright information, and REM statements may be eliminated completely. The ten year results may also be deleted. Many statements may be combined, especially those referring to the same line. In general, the program must be streamlined to fit the ZX-80.

Prizes

1st—A one year subscription or renewal to SYNC, and a SYNC T shirt.
2nd—A one year subscription or renewal to SYNC.
3rd—A SYNC T shirt.

Send all entries to:
The SYNC Challenge
39 East Hanover Avenue
Morris Plains, New Jersey 07950

Hammurabi

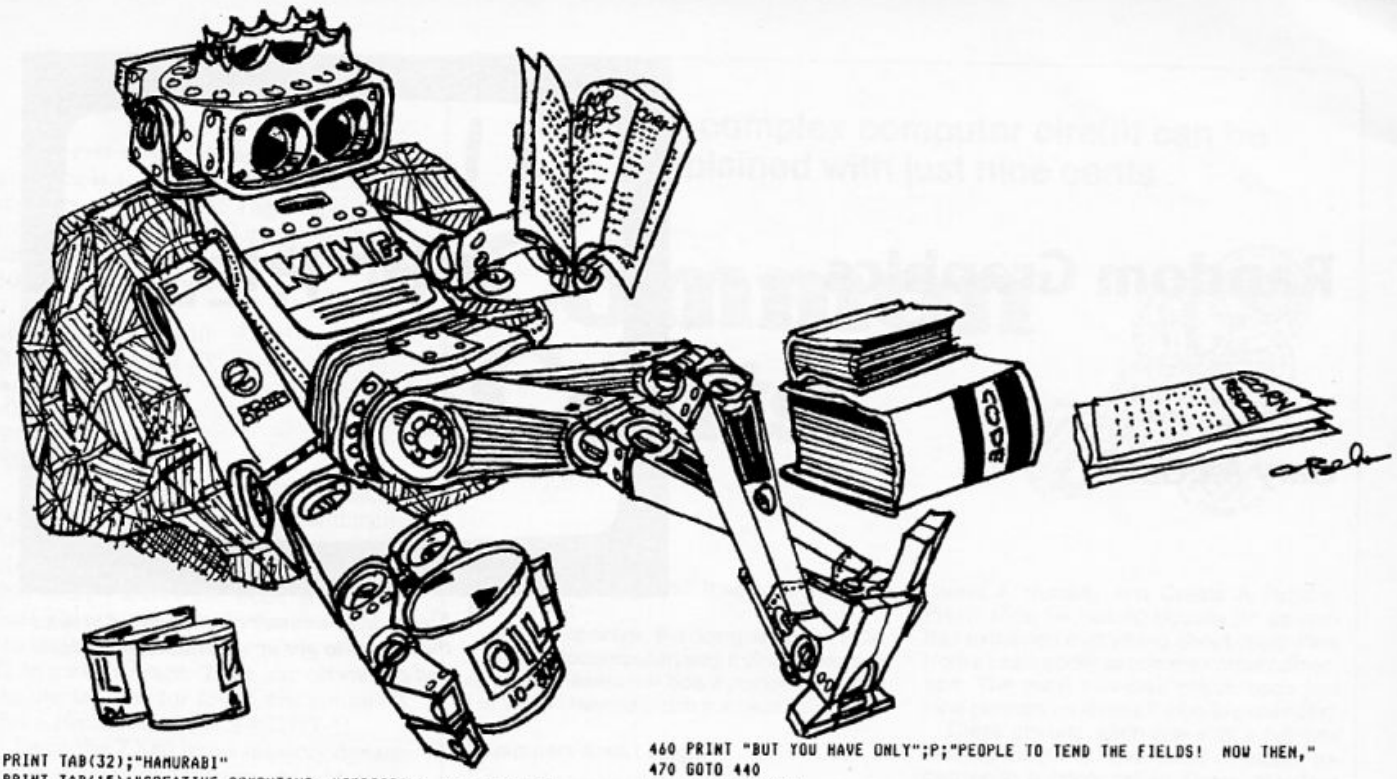
In this game you direct the administrator of Sumeria, Hammurabi, how to manage the city. The city initially has 1,000 acres, 100 people and 3,000 bushels of grain in storage.

You may buy and sell land with your neighboring city-states for bushels of grain — the price will vary between 17 and 26 bushels per acre. You also must use grain to feed your people and as seed to plant the next year's crop.

You will quickly find that a certain number of people can only tend a certain amount of land and that people starve if they are not fed enough. You also have the unexpected to contend with such as a plague, rats destroying stored grain, and variable harvests.

You will also find that managing just the few resources in this game is not a trivial job over a period of say ten years. The crisis of population density rears its head very rapidly.

```
                HAMURABI
CREATIVE COMPUTING  MORRISTOWN, NEW JERSEY

TRY YOUR HAND AT GOVERNING ANCIENT SUMERIA
FOR A TEN-YEAR TERM OF OFFICE.

HAMURABI:  I BEG TO REPORT TO YOU,
IN YEAR 1 , 0 PEOPLE STARVED, 5 CAME TO THE CITY,
POPULATION IS NOW 100
THE CITY NOW OWNS 1000 ACRES.
YOU HARVESTED 3 BUSHELS PER ACRE.
RATS ATE 200 BUSHELS.
YOU NOW HAVE 2800 BUSHELS IN STORE.

LAND IS TRADING AT 24 BUSHELS PER ACRE.
HOW MANY ACRES DO YOU WISH TO BUY? 10

HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE? 2000

HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED? 990

HAMURABI:  I BEG TO REPORT TO YOU,
IN YEAR 2 , 0 PEOPLE STARVED, 5 CAME TO THE CITY,
POPULATION IS NOW 105
THE CITY NOW OWNS 1010 ACRES.
YOU HARVESTED 3 BUSHELS PER ACRE.
RATS ATE 16 BUSHELS.
YOU NOW HAVE 3019 BUSHELS IN STORE.

LAND IS TRADING AT 21 BUSHELS PER ACRE.
HOW MANY ACRES DO YOU WISH TO BUY? 25

HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE? 2000

HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED? 1000
HAMURABI:  THINK AGAIN.  YOU HAVE ONLY
 494 BUSHELS OF GRAIN.  NOW THEN,
HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED? 500

HAMURABI:  I BEG TO REPORT TO YOU,
IN YEAR 3 , 5 PEOPLE STARVED, 5 CAME TO THE CITY,
A HORRIBLE PLAGUE STRUCK!  HALF THE PEOPLE DIED.
POPULATION IS NOW 52
THE CITY NOW OWNS 1035 ACRES.
YOU HARVESTED 1 BUSHELS PER ACRE.
RATS ATE 0 BUSHELS.
YOU NOW HAVE 744 BUSHELS IN STORE.

LAND IS TRADING AT 17 BUSHELS PER ACRE.
HOW MANY ACRES DO YOU WISH TO BUY? 0
HOW MANY ACRES DO YOU WISH TO SELL? 25

HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE? 1000

HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED? 500
HAMURABI:  THINK AGAIN.  YOU HAVE ONLY
 169 BUSHELS OF GRAIN.  NOW THEN,
HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED? 300


HAMURABI:  I BEG TO REPORT TO YOU,
IN YEAR 4 , 2 PEOPLE STARVED, 12 CAME TO THE CITY,
POPULATION IS NOW 62
THE CITY NOW OWNS 1010 ACRES.
YOU HARVESTED 1 BUSHELS PER ACRE.
RATS ATE 0 BUSHELS.
YOU NOW HAVE 319 BUSHELS IN STORE.

LAND IS TRADING AT 23 BUSHELS PER ACRE.
HOW MANY ACRES DO YOU WISH TO BUY? 0
HOW MANY ACRES DO YOU WISH TO SELL? 500

HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE? 500

HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED? 230

YOU STARVED 37 PEOPLE IN ONE YEAR!!!
DUE TO THIS EXTREME MISMANAGEMENT YOU HAVE NOT ONLY
BEEN IMPEACHED AND THROWN OUT OF OFFICE BUT YOU HAVE
ALSO BEEN DECLARED NATIONAL FINK!!!!

SO LONG FOR NOW.
```




```
10 PRINT TAB(32);"HAMURABI"
20 PRINT TAB(15);"CREATIVE COMPUTING  MORRISTOWN, NEW JERSEY"
30 PRINT:PRINT:PRINT
80 PRINT "TRY YOUR HAND AT GOVERNING ANCIENT SUMERIA"
90 PRINT "FOR A TEN-YEAR TERM OF OFFICE.":PRINT
95 D1=0: P1=0
100 Z=0: P=95:S=2800: H=3000: E=H-S
110 Y=3: A=H/Y: I=5: Q=1
210 D=0
215 PRINT:PRINT:PRINT "HAMURABI:  I BEG TO REPORT TO YOU,": Z=Z+1
217 PRINT "IN YEAR";Z;",";D;"PEOPLE STARVED,";I;"CAME TO THE CITY,"
218 P=P+I
227 IF Q>0 THEN 230
228 P=INT(P/2)
229 PRINT "A HORRIBLE PLAGUE STRUCK!  HALF THE PEOPLE DIED."
230 PRINT "POPULATION IS NOW";P
232 PRINT "THE CITY NOW OWNS";A;"ACRES."
235 PRINT "YOU HARVESTED";Y;"BUSHELS PER ACRE."
250 PRINT "RATS ATE";E;"BUSHELS."
260 PRINT "YOU NOW HAVE";S;"BUSHELS IN STORE.": PRINT
270 IF Z=11 THEN 860
310 C=INT(10*RND(1)): Y=C+17
312 PRINT "LAND IS TRADING AT";Y;"BUSHELS PER ACRE."
320 PRINT "HOW MANY ACRES DO YOU WISH TO BUY";
321 INPUT Q: IF Q<0 THEN 850
322 IF Y*Q<=S THEN 330
323 GOSUB 710
324 GOTO 320
330 IF Q=0 THEN 340
331 A=A+Q: S=S-Y*Q: C=0
334 GOTO 400
340 PRINT "HOW MANY ACRES DO YOU WISH TO SELL";
341 INPUT Q: IF Q<0 THEN 850
342 IF Q<A THEN 350
343 GOSUB 720
344 GOTO 340
350 A=A-Q: S=S+Y*Q: C=0
400 PRINT
410 PRINT "HOW MANY BUSHELS DO YOU WISH TO FEED YOUR PEOPLE";
411 INPUT Q
412 IF Q<0 THEN 850
418 REM *** TRYING TO USE MORE GRAIN THAN IS IN SILOS?
420 IF Q<=S THEN 430
421 GOSUB 710
422 GOTO 410
430 S=S-Q: C=1: PRINT
440 PRINT "HOW MANY ACRES DO YOU WISH TO PLANT WITH SEED";
441 INPUT D: IF D=0 THEN 511
442 IF D<0 THEN 850
444 REM *** TRYING TO PLANT MORE ACRES THAN YOU OWN?
445 IF D<=A THEN 450
446 GOSUB 720
447 GOTO 440
449 REM *** ENOUGH GRAIN FOR SEED?
450 IF INT(D/2)<=S THEN 455
452 GOSUB 710
453 GOTO 440
454 REM *** ENOUGH PEOPLE TO TEND THE CROPS?
455 IF D<10*P THEN 510
460 PRINT "BUT YOU HAVE ONLY";P;"PEOPLE TO TEND THE FIELDS!  NOW THEN,"
470 GOTO 440
510 S=S-INT(D/2)
511 GOSUB 800
512 REM *** A BOUNTIFUL HARVEST!
515 Y=C: H=D*Y: E=0
521 GOSUB 800
522 IF INT(C/2)<>C/2 THEN 530
523 REM *** RATS ARE RUNNING WILD!!
525 E=INT(S/C)
530 S=S-E+H
531 GOSUB 800
532 REM *** LET'S HAVE SOME BABIES
533 I=INT(C*(20*A+S)/P/100+1)
539 REM *** HOW MANY PEOPLE HAD FULL TUMMIES?
540 C=INT(Q/20)
541 REM *** HORROR, A 15% CHANCE OF PLAGUE
542 Q=INT(10*(2*RND(1)-.3))
550 IF P<C THEN 210
551 REM *** STARVE ENOUGH FOR IMPEACHMENT?
552 D=P-C: IF D>.45*P THEN 560
553 P1=((Z-1)*P1+D*100/P)/Z
555 P=C: D1=D1+D: GOTO 215
560 PRINT: PRINT "YOU STARVED";D;"PEOPLE IN ONE YEAR!!!"
565 PRINT "DUE TO THIS EXTREME MISMANAGEMENT YOU HAVE NOT ONLY"
566 PRINT "BEEN IMPEACHED AND THROWN OUT OF OFFICE BUT YOU HAVE"
567 PRINT "ALSO BEEN DECLARED NATIONAL FINK!!!!": GOTO 990
710 PRINT "HAMURABI:  THINK AGAIN.  YOU HAVE ONLY"
711 PRINT S;"BUSHELS OF GRAIN.  NOW THEN,"
712 RETURN
720 PRINT "HAMURABI:  THINK AGAIN.  YOU OWN ONLY";A;"ACRES.  NOW THEN,"
730 RETURN
800 C=INT(RND(1)*5)+1
801 RETURN
850 PRINT: PRINT "HAMURABI:  I CANNOT DO WHAT YOU WISH."
855 PRINT "GET YOURSELF ANOTHER STEWARD!!!!!"
857 GOTO 990
860 PRINT "IN YOUR 10-YEAR TERM OF OFFICE,";P1;"PERCENT OF THE"
862 PRINT "POPULATION STARVED PER YEAR ON THE AVERAGE, I.E. A TOTAL OF"
865 PRINT D1;"PEOPLE DIED!!": L=A/P
870 PRINT "YOU STARTED WITH 10 ACRES PER PERSON AND ENDED WITH"
875 PRINT L;"ACRES PER PERSON.": PRINT
880 IF P1>33 THEN 565
885 IF L<7 THEN 565
890 IF P1>10 THEN 940
892 IF L<9 THEN 940
895 IF P1>3 THEN 960
896 IF L<10 THEN 960
900 PRINT "A FANTASTIC PERFORMANCE!!!  CHARLEMAGNE, DISRAELI, AND"
905 PRINT "JEFFERSON COMBINED COULD NOT HAVE DONE BETTER!":GOTO 990
940 PRINT "YOUR HEAVY-HANDED PERFORMANCE SMACKS OF NERO AND IVAN IV."
945 PRINT "THE PEOPLE (REMAINING) FIND YOU AN UNPLEASANT RULER, AND,"
950 PRINT "FRANKLY, HATE YOUR GUTS!!":GOTO 990
960 PRINT "YOUR PERFORMANCE COULD HAVE BEEN SOMEWHAT BETTER, BUT"
965 PRINT "REALLY WASN'T TOO BAD AT ALL. ";INT(P*.8*RND(1));"PEOPLE"
970 PRINT "DEARLY LIKE TO SEE YOU ASSASSINATED BUT WE ALL HAVE OUR"
975 PRINT "TRIVIAL PROBLEMS."
990 PRINT: FOR N=1 TO 10: PRINT CHR$(7);: NEXT N
995 PRINT "SO LONG FOR NOW.": PRINT
999 END
```

Random Graphics

Gary McGath

Figure 1. The cursor can be moved around the screen to put up entertaining diagrams and pictures.

A large part of the fun of having a home computer is being able to put pictures up on the display screen. This can be done on the ZX80, thanks to the graphics characters. You simply have to PRINT as many lines of the appropriate characters as you want. These characters can be generated one at a time, using the statement

PRINT CHR$(N);

with appropriate values of N.

So you write your program, putting up as much as twenty-four rows of thirty-two characters; you run it, and in a rather large fraction of a minute, you have your picture.

Fine. But now, suppose you want to make a small change in the picture in response to an input. You have to clear the screen and do the display all over, taking another large chunk of a minute. Having to redo the whole screen for every change can get tedious, to say nothing of making the programming difficult.

But it is possible to do better. Appendix III of the ZX80 Operating Manual tells us the location of system variables in memory. At location 16396 we find a variable called D FILE, which contains the address of the beginning of the display buffer. We can get this address by taking PEEK (16396)+256*PEEK(16397). By doing POKEs at offsets from this address, we can manipulate individual characters at random without disturbing the rest of the display. This random access technique allows much faster small changes to what you see on the screen.

Gary McGath, 5 Ames Rd., RFD 3, Milford, NH 03055.

```
 10 FOR I=1 TO 20
 20 FOR J=1 TO 30
 30 PRINT " ";
 40 NEXT J
 50 PRINT
 60 NEXT I
100 LET X=1
110 LET Y=1
120 LET P=X+31*(Y-1)
130 POKE PEEK (16396) + 256*PEEK (16397) +P, 141
140 INPUT X$
145 IF X$="" THEN STOP
150 POKE PEEK (16396) +256*PEEK (16397) +P, CODE (X$)
160 LET Y$=TL$ (X$)
170 IF Y$= "R" AND X<30 THEN LET X=X+1
180 IF Y$="L" AND X>1 THEN LET X=X-1
190 IF Y$="D" AND Y<20 THEN LET Y=Y+1
200 IF Y$="U" AND Y>1 THEN LET Y=Y-1
210 GO TO 120
```

The first step in doing random-access graphics is to give yourself some screen memory to play with. The simplest way to do this is by PRINTing a number of lines of equal length. For example, to do 10 rows of 20 characters:

```
10  FOR I=1 TO 10
20  FOR J=1 TO 20
30  PRINT " ";
40  NEXT J
50  PRINT
60  NEXT I
```

You can now address specific characters in the display with some simple arithmetic. Remember that the first character in the display is always in a Newline code, and that there's a Newline code after every line. This means that if your lines are 20 characters long, each line really takes up 21 bytes of storage. Thus, the offset into the display area for row Y and column X (for a 10x20 display) is X=21*(Y-1).

Since the ZX80 treats memory dynamically, moving things from one address to another in unexpected ways, you should POKE screen locations in the same statement that contains the PEEK to obtain the screen address. You can't just PEEK once to find out where the display start and assume it will stay there ever after. So, to put the character CHR$(C) at row Y and column X, the necessary statements are:

```
LET P = X+21*(Y-1)
POKE PEEK(16396)+256*PEEK(16397)
+P,C
```

This is still assuming 20-character lines.

You can do INPUTs at the bottom of the screen while playing with the display in this manner. This lets you enter commands to manipulate the display.

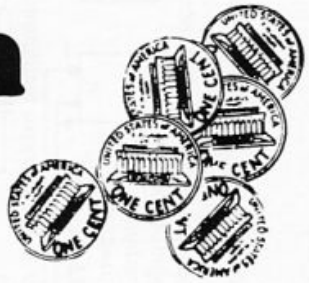
The program in Figure 1 lets you move a cursor around the screen and put graphics up where the cursor is. You operate the program as follows:

The required input is a two-character string. The first character gets displayed where the cursor (an inverted dollar sign) is; the second character, which must be L, R, U, or D, tells the cursor which way to move (left, right, up, or down). To exit from the program type a null string (just Newline).

A word of caution is necessary; Since you're doing PEEKs and POKEs, errors in random graphics programs can easily crash the ZX80. If this happens, just unplug it for a moment and try again. This will, of course, wipe out the program in memory, so you might consider saving it on cassette *before* testing it. It's worth the extra difficulty, though, since this technique lets you do things that never seemed possible before on the ZX80's display.

The most complex computer circuit can be explained with just nine cents

Common Cents



The "penny switch." It sounds strange. But it's not.

Joe Weisbecker, the designer of the RCA 1802 microcomputer, was trying to explain to some children just how a computer works. He wasn't having much success.

Computers Aren't Magic

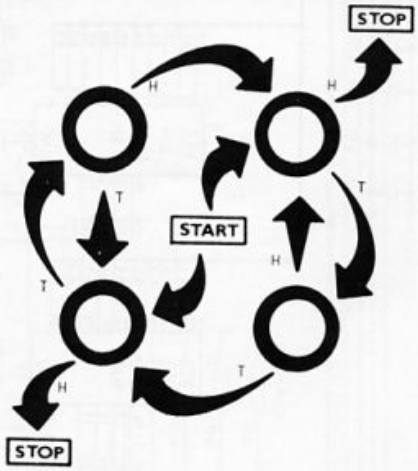
Joe's hobby is magic. He thought, "maybe I can use some kind of illusion to show how a computer works." But he didn't really want to use an illusion. He didn't want the children to think of a computer as magic.

So he hit upon the idea of a simple flip-flop switch (the most common circuit in a computer) represented by the head or tail of a penny. This flip-flop circuit uses just one penny. Every time it receives an impulse it changes from head to tail or tail to head. Simple.

But then Joe went on and put two of these simple flip flops together to make a circuit that adds two numbers together. And another that subtracts numbers. Kids loved these circuits and played with them like games.

Games With Pennies

Before long, Joe devised circuits to play more complicated games like Tic Tac Toe, Guess A Number and Create A Pattern. Pretty soon he had 30 circuits (or games) that explained everything about computers from a basic adder to complex error correction. The most complex circuit uses just nine pennies (or dimes for the big spender).



"Heads Up Game." Starting with tails in all positions, how many times through to get all four pennies heads up?

These circuits, each one with a full size playing diagram, have been collected together in a book called *Computer Coin Games*. With this book children or adults can easily understand the workings of even the most complex computer circuits.

Games Magazine said, "whether or not you have any experience with computer technology, you'll be both amazed and delighted with the simplicity of the format and the complexity of the play. All you need is some common cents."

Dr. Dobbs Journal agreed, saying, "*Computer Coin Games* is a simple approach to a complicated concept. The book is liberally sprinkled with clever illustrations and diagrams, and provides a relatively painless route to understanding how computer circuits function."

Money back Guarantee

We're convinced that you'll understand the inner workings of a computer after playing these 30 games. If you don't, send the book back and we'll refund the complete price plus your postage to send it back.

To order your copy of *Computer Coin Games*, just send $3.95 plus $1.00 shipping and handling to Creative Computing Press, Morris Plains, NJ 07950. Visa, MasterCard and American Express orders may be called toll free to 800-631-8112 (in NJ, 201-540-0445).

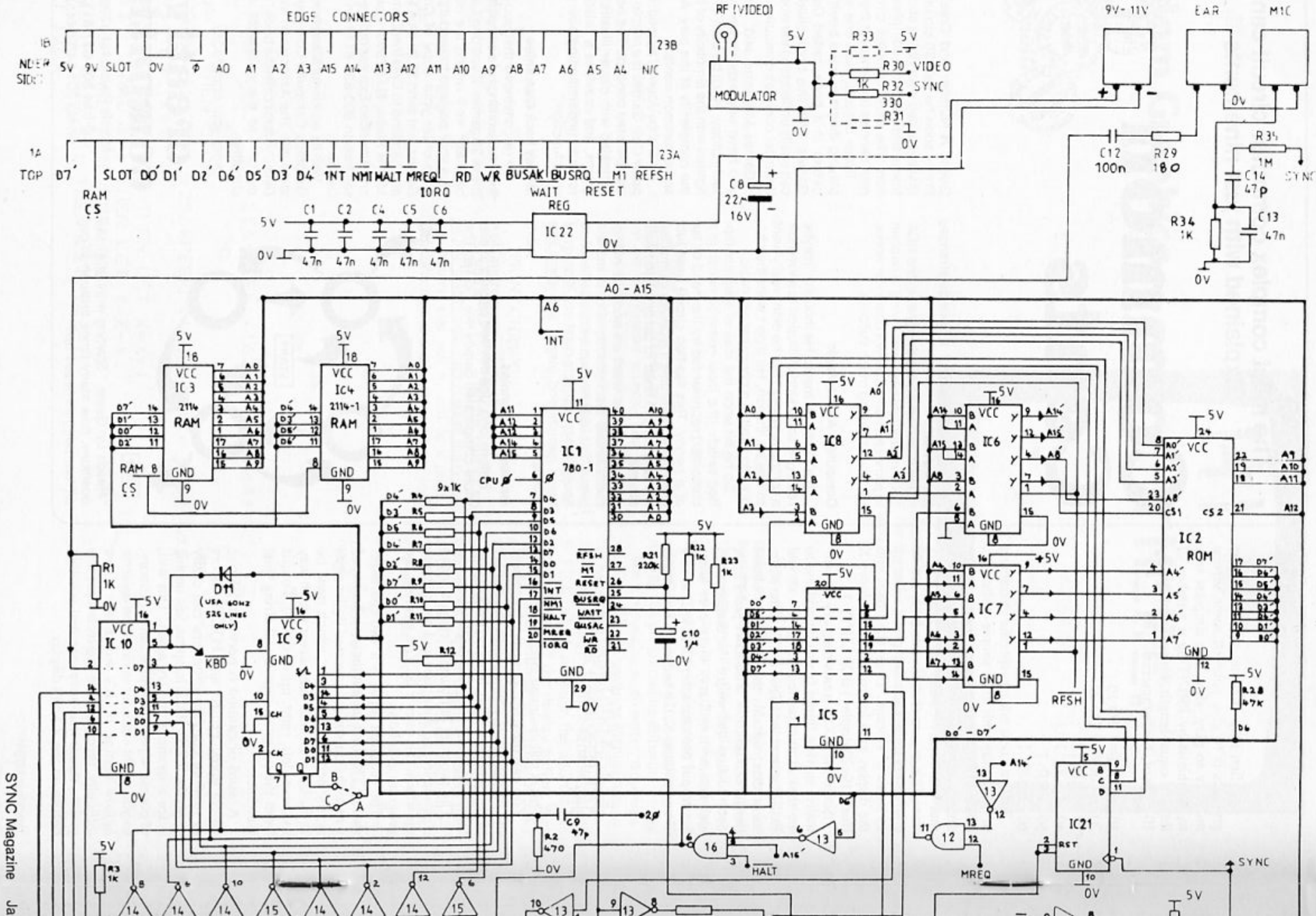
With its wonderful illustrations by Sunstone Graphics, *Computer Coin Games* makes an ideal gift. *The Association for Educational Data Systems* calls the book "an ideal introduction to the concepts of computer circuitry."

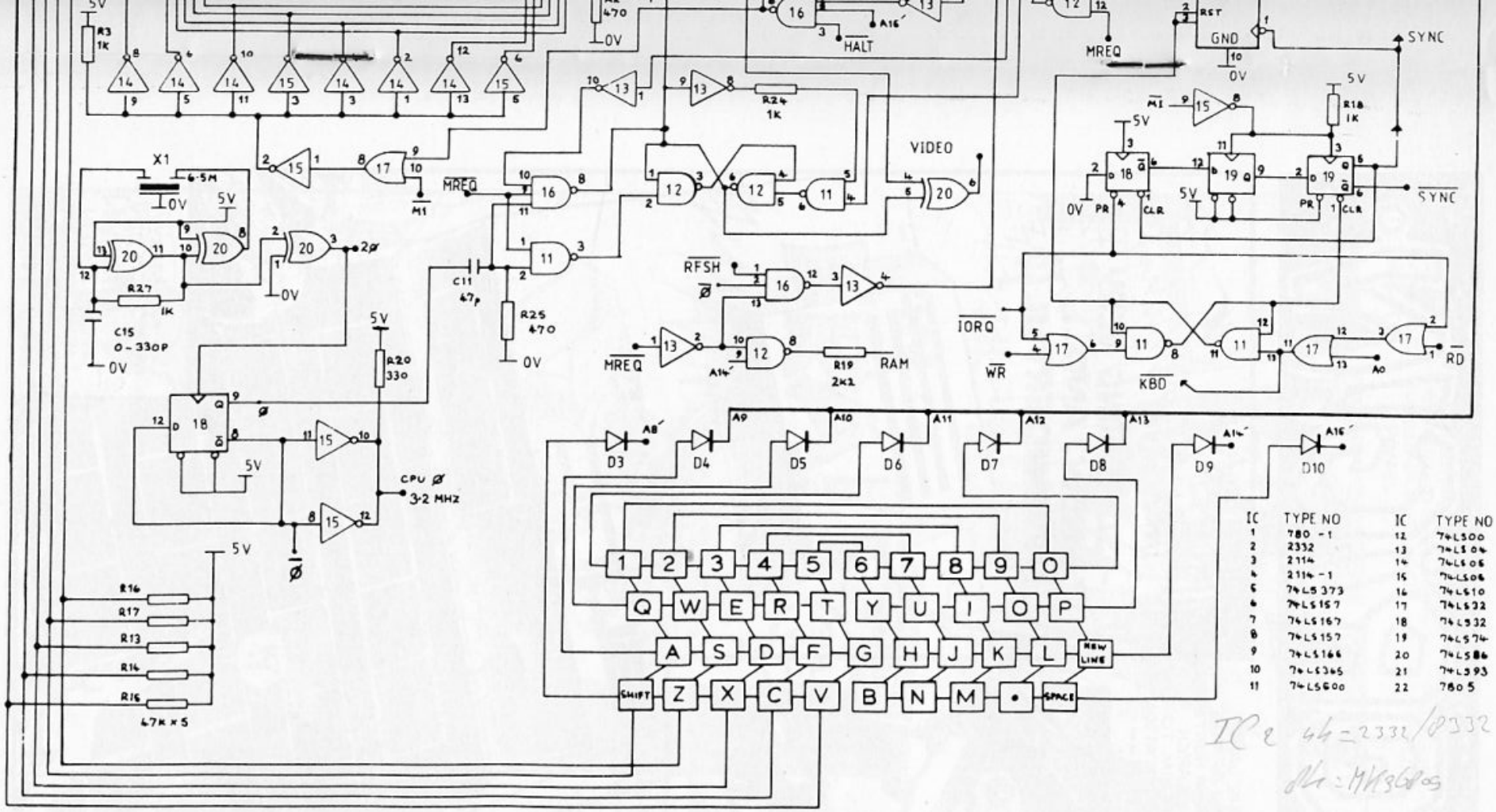
Order your copy today.

creative computing

Morris Plains, NJ 07950
Toll-free **800-631-8112**
(In NJ 201-540-0445)

ZX80 Circuit Diagram





IC	TYPE NO	IC	TYPE NO
1	780 - 1	12	74LS00
2	2332	13	74LS04
3	2114	14	74LS05
4	2114 - 1	15	74LS06
5	74LS373	16	74LS10
6	74LS157	17	74LS32
7	74LS167	18	74LS32
8	74LS157	19	74LS74
9	74LS166	20	74LS86
10	74LS365	21	74LS93
11	74LS600	22	7805

IC2 44=2332/8332
Ph=MK3G809

ZX-80 Technical Fact Sheet

CPU
Type: Z80A (or NEC equivalent.)
Speed: 3.25 MHz

ROM
Type: 8332 (Mask programmed by Electronic Arrays)
Size: 4K x 8
Replacements-

Type	Size
2716	2K x 8
2732	4K x 8

RAM
Type: 2114
Size: 1K x 4
Speed: 450ns

POWER
Input: 9-11 Vdc @ 350-500 ma (Typ. 400ma)
Regulator: 7805 (Internal. Heatsinked.)
Filtered: Yes
Despiked: Yes

MOUNTING
CPU, RAM and ROM are socketed. All other chips are soldered to board.

PCB
Type: Double sided. Plated through.

CASE
Size: 8 1/2" X 6 1/2" x 1 1/2"

KEYBOARD:
Type: 40 key touch-sensitive membrane.

INPUT/OUTPUT
A.) 3.5 mm Power Jack
B.) 3.5 mm Earphone Jack
C.) 3.5 mm Microphone Jack
D.) 46 pin polarized edge connector (accesses all Z80A lines)
E.) R.F. Video output @ VHF channel 2

CORDS
A.) 4' Video Cord
B.) Split double 2 conductor cassette cord

ORIGIN:
THE WORLD IS VERY YOUNG-- A PLACE WHERE ONLY THE VERY STRONG...OR THE VERY WISE... SURVIVE...!
IT IS THE LAND AND TIME OF THE HUNTER--A HUNTER WHOSE SEARCH HAS LED HIM TO A CONFRONTATION WITH--
--SOMETHING VERY NEW, AND... WONDERFUL!!
STUPIFIED, THE HUNTER GAZES UP AT THIS NEW MARVEL, HIS THOUGHTS ASKING QUESTIONS HIS PRIMITIVE BRAIN CAN'T ANSWER...!!
!?
?!
SYNC
ART-STORY/ ©1980 TIMOTHY TRUMAN
EDITOR/ J. STEIN
MUSIC/ THE LONDON PHILHARMONIC ORCHESTRA

THE HUNTER CAUTIOUSLY APPROACHES THE STRANGE MONOLITH...!

UPWARDS HE STRETCHES, FASCINATED, REACHING, TO TOUCH THE SMOOTH, WASHABLE, PLASTIC PANEL...!

HIS NERVOUSLY GROPING FINGERS TOUCH THE COOL, PRESSURE-SENSITIVE SYMBOLS...

...TO BEGIN...

--AN ERA!

HE IS NOW A MODERN-MAN... A MAN WITH PLACES TO GO:

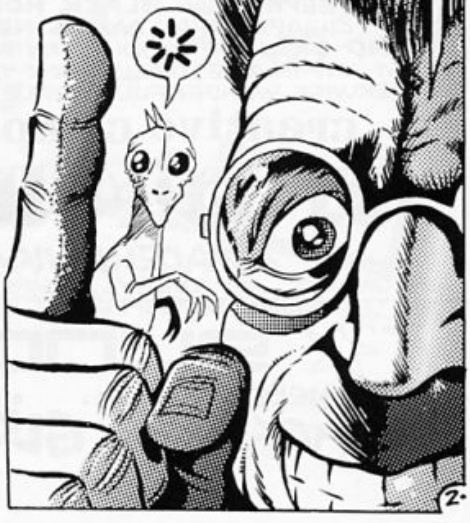
DAYTON! NIAGARA FALLS! McDONALDS!

...PEOPLE TO SEE:

CHARLIE'S ANGELS! THE N.Y. YANKEES! BARRY MANILOW!

...THINGS TO DO:

MOW THE LAWN! JOIN THE BOWLING LEAGUE! CHANGE THE KITTY-LITTER! LUBE THE CAR! PICK UP THE KIDS! CLEAN THE STOVE! PICK LINT FROM THE DRIER FILTER! CHRISTMAS SHOP!



2.

MORE SPECTACULAR THAN **STAR WARS!**
FLASHIER THAN **FLASH GORDON!**
SEXIER THAN **SATURN 3!**
DEEPER THAN **BLACK HOLE!**
CHEAPER THAN **MARS NEEDS WOMEN!**
...**AND** BROUGHT TO YOU AT AN INCREDIBLY LOW COST AND IN AWESOMELY POOR TASTE BY THE OTHERWISE-WONDERFUL PEOPLE WHO GIVE YOU

creative computing!

IT'S...

crash cursor!

-SPACEHAWK!

IN THE NEXT... AND **EVERY!**... ISSUE OF THE DARING, DYNAMIC, AND DIFFERENT NEW...

SYNC

JOIN US **NEXT ISSUE** FOR:

"ATTACK OF THE GLITCHOIDZ!"

A one-hour LP record of eight synthesizers may change your views about computer music forever

Binary Beatles

by David Ahl

Computer music. Who needs it? It's mostly boring beep, beep, beeps or wildly modern stuff. It's certainly nothing you'd want to listen to more than once. That's what I thought about computer music and most of my friends agreed.

In 1978 I entered Yankee Doodle Dandy into my Software Technology system just to be different. Dick Moberg heard of it and asked me to perform in the Philadelphia Computer Music Festival. I agreed expecting to be the only one with something out of the ordinary. I was wrong.

Computer Accompanist

Nine individuals and groups performed in the festival. There were the usual Bach pieces but even they were different. Gooitzen van der Wal performed the last movement of the 2nd Bach Suite in a unique way. He played the flute solo while using the computer as accompaniment.

Then Dorothy Siegel did the same thing, playing the clarinet solo part of Wanhal's Sonata in b flat. The audience went wild.

Hal Chamberlin played Bach's Tocatta and Fugue in d minor. But also with a difference. He used a large computer before hand to "compute" the waveform of every instrument playing every note. It took one hour of computation time for each two minutes of playback time. The result could hardly be distinguished from the organ in the Hapsburg Cathedral.

Don Schertz had a home brewed synthesizer truly mounted on a breadboard that allowed him to control 25 parameters of each note. It produced spectacular sounds in his arrangement of Red Wing.

Singing Computer

In 1962, D.H. Van Lenten at Bell Laboratories produced the first talking computer. Bell engineers taught it to recite the soliloquy from Hamlet. Then they went one step further and taught it to sing Daisy both alone and accompanied by another computer. This was also performed at the festival.

Yes, the Beatles were represented. Andrew Molda played Hey Jude on his COSMAC VIP system with a program called PIN-8 (Play it Now).

Superb Quality Recording

All these pieces and twelve others were recorded with broadcast quality equipment. Because of audience noise, eight were re-recorded later in a studio. We then took these tapes to Tru-Tone, a top recording studio and cut a lacquer master. It was a long session since the recording engineers insisted upon analyzing the sound from every source and setting up the equilization curves accordingly. It took over 12 hours to produce a one-hour lacquer master.

Finished recordings were then pressed on top-quality vinyl and inserted into liners and record jackets. These were then shrink wrapped in plastic for maximum protection. We guaranteee that every LP record is free from defects or we will replace it free of charge.

The extensive descriptions of each of the eight synthesizers and the festival would not all fit on the jacket so we've included an extra sheet with each record. This entire package is mailed in a protective corrugated package to insure that it reaches you in mint condition. The cost is a modest $6.00 postpaid in the U.S. and $7.00 foreign. Send order with payment or Visa, MasterCard or American Express number to Creative Computing, Morris Plains, NJ 07950.

This LP record contains one hour of eight computer music synthesizers that you'll listen to over and over again. Send in your order today at no risk whatsoever.

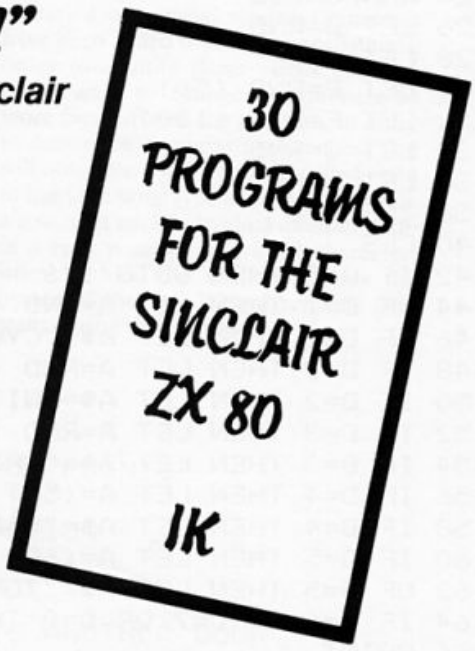
creative computing

Morris Plains, NJ 07950
Toll-free **800-631-8112**
(In NJ 201-540-0445)

"This Book is Excellent!"

— Clive Sinclair

The unique book contains 30 programs all designed to fit in the basic 1K version of the SINCLAIR ZX80!!
With this book you will realise that the ZX80 is more powerful than you ever imagined!
112 pages packed full of solid information!
BLACKJACK — actually contains a full pack of cards, shuffles them, keeps track of the dealer and player totals, and the money bets, all within 1K.
Dr. ZX-80 — a truly conversational program: DR. ZX-80 is your personal computer analyst.
LINE RENUMBER — an invaluable program which automatically renumbers lines and puts order to your programs.
MEMORY LEFT — an incredible routine especially useful with only 1K, which lets you know to the byte how much memory is left. This also illustrates USR routines.
GOMOKU — the computer challenges you to this complex Japanese game, incredibly this program including display of the 7 x7 board fits into 1K — it only does so because it uses the display as memory!
Other programs included are **HORSE RACE, LUNAR LANDER** (with moving spaceship display), **NOUGHTS AND CROSSES, NIM, SIMPLE SIMON, HANGMAN, LIFE: MASTERMIND, PINCH** and 16 others!



available by mail order only

$14.95
(plus $1 shipping and handling)

Other books in preparation include:
ZX80 Machine Language Programming

MELBOURNE HOUSE PUBLISHERS

Orders to: **IMAGE COMPUTER PRODUCTS**
615 Academy Drive, Northbrook, Ill. 60062.

Please send me _____ copy/ies of '**30 Programs for the ZX-80: 1K**

NAME _____

ADDRESS _____

_____

Please encolse
Check or Money Order
for $15.95

Castle Doors 2K

Weldon J. Horton Ph.D

```
 1 PRINT
 2 PRINT
 4 PRINT "WELCOME TO CASTLE DOORS"
 6 PRINT
 8 LET B=RND (300)  --Wizard's strength--
10 LET C=RND (50)   --Starting strength of player--
12 PRINT
14 PRINT
16 PRINT "YOUR STRENGTH IS ";C
18 PRINT
20 PRINT
22 PRINT "PRESS 4 TO OPEN A DOOR, PRESS 5 TO FIGHT THE WIZARD"
24 RANDOMISE
26 LET D=RND (8)           --Door selector--
28 LET A=0                 --Initializes monster strength--
30 LET E=RND (50)          --Size of acceptable bribe--
32 LET F=RND (100)         --Chance of getting caught on run option
34 LET G=RND (25)          --Gain in strength on win over monster--
36 LET H=RND (15)          --Loss of strength on run option--
38 INPUT O
40 CLS
42 IF O=5 THEN GOTO 176
44 IF D=1 THEN LET A=RND (20)   --Strength of monster in line below--
46 IF D=1 THEN LET A$="CYCLOPS"     --Monster name--
48 IF D=2 THEN LET A=RND (30)
50 IF D=2 THEN LET A$="WITCH"
52 IF D=3 THEN LET A=RND (40)
54 IF D=3 THEN LET A$="DRAGON"
56 IF D=4 THEN LET A=(50)
58 IF D=4 THEN LET A$="GARGOYLE"
60 IF D=5 THEN LET A=(60)
62 UF D=5 THEN LET A$="ZOMBIE"
64 IF D=6 OR D=7 OR D=8 THEN GOTO 160
66 PRINT
68 PRINT
70 PRINT "YOUR OPPONENT IS A ";A$
72 PRINT
74 PRINT
76 PRINT "YOUR STRENGTH IS ";C
78 PRINT
```

Castle doors is an adventure game that presents a new challenge with each playing, and the opportunity for easy adjustment of difficulty if the Wizard is winning too much, or too little of the time.

You, the player, have the ultimate task of defeating the evil Wizard, but first you must gain strength and experience by confronting the strange denizens of the Wizard's enchanted castle. Behind each door of the castle lies a new adventure. Some of the castle dwellers are friendly, (like the Fairy, who gives you a magic potion that doubles your strength) but

Weldon J. Horton, 3215 W. Degan, Midland, TX 79701.

```
 80 PRINT
 82 PRINT "YOU MAY: FIGHT=1 BRIBE=2 RUN=3"
 84 INPUT N
 86 CLS
 88 IF N=2 THEN GOTO 124
 90 IF N=3 THEN GOTO 148
 92 IF C<A THEN GOTO 176
 94 PRINT
 96 PRINT
 98 PRINT "YOUR STRENGTH IS ";C
100 PRINT
102 PRINT
104 PRINT "HOW MUCH STRENGTH DO YOU WANT
    TO FIGHT WITH?"
106 INPUT Q
108 CLS
109 PRINT
110 PRINT
112 IF Q>C THEN GOTO 94
114 IF Q>A OR Q=A THEN LET C=C+G
116 IF Q>A OR Q=A THEN PRINT "YOU WIN
    BRAVE ADVENTURER"
118 IF Q<A THEN LET C=C-Q
120 IF Q<A THEN PRINT "YOU LOSE TO THE ";A$
122 GOTO 12
124 PRINT
126 PRINT
128 PRINT "YOUR STRENGTH IS ";C
130 PRINT
132 PRINT "HOW BIG A BRIBE DO YOU
    WANT TO OFFER?"
134 INPUT P
136 CLS
138 IF P>E OR P=E THEN LET C=C-P+G
140 IF P>E OR P=E THEN PRINT "BRIBE ACCEPTED"
142 IF P<E THEN PRINT "BRIBE REFUSED"
144 IF P<E THEN GOTO 66
146 GOTO 12
148 PRINT
150 PRINT
152 IF F=50 THEN GOTO 176
154 PRINT "YOU GOT AWAY FROM THE MONSTER"
156 LET C=C-H
158 GOTO 12
160 PRINT
162 PRINT
164 IF D=6 THEN PRINT "SAVE A PRINCESS AND GAIN 50 POINTS"
166 IF D=6 THEN LET C=C+50
168 IF D=7 THEN PRINT "FAIRY GIVES YOU A MAGIC POTION
    THAT DOUBLES YOU STRENGTH"
170 IF D=7 THEN LET C=C*2
172 IF D=8 THEN PRINT "A GIANT BAT CARRIES YOU TO ANOTHER DOOR"
174 GOTO 12
176 PRINT
178 PRINT
180 IF C>B OR C=B THEN PRINT "THE WIZARD IS DEFEATED-YOU WIN"
182 IF C<B OR C<A OR F=50 THEN PRINT "YOU FOUGHT BRAVELY,
    BUT WERE DEFEATED-GAME OVER"
184 STOP
```

most offer you only of of three alternatives: fight the monster and pass through the door, bribe the monster to let you pass, or you can run away from the monster.

After chosing to fight the monster, you must decide how much strength to fight with. If you lose, you lose the strength you fought with; if you win, you win additional strength. If your total strength is less than the strength of the monster whom you are fighting, you lose the game.

If you chose to bribe the monster to let you pass, it may refuse your bribe, and you must decide what to do next: fight, offer a bigger bribe, or run away. If your bribe is accepted you lose the bribe and gain the same amount of strength points you would have had if you had fought the monster and won. Be warned, however, that a successful bribe can easily result in an overall loss of points.

If you chose to run away you will, naturally, lose some strength. Also, the monster has a one in one-hundred chance of catching you and ending the game.

When you feel that you are ready, you may chose to fight the evil Wizard. The battle is short, fierce, and has the immediate outcome of either total victory or defeat.

This program is written in such a manner as to make the adjustment of all strength and penalty values quite easy. Messages printed in the course of the program run have been adjusted to print correctly on the screen by the addition or delection of internal spaces. Comments have been added to the program listing in order to identify those values that you might want to change as you become more familiar with the program, in order to customize it to your preferences. You will note, that even if you are called upon to fight the same type of monster twice in a row, that each individual monster within a type posesses individual characteristics, withing the set limits. Only the power of the evil Wizard remains constant during each game.

Draw a Picture

Howard Branch

```
100 PRINT
110 PRINT
120 PRINT "SIZE PAD (3 - 7)?"
130 INPUT N
140 IF N < 3 OR N > 7 THEN TO TO 120
150 DIM A (N * N - 1)
200 CLS
210 FOR K = 1 TO N
220 LET J = N + 1 - K
230 PRINT J,
240 FOR I = 1 TO N
250 PRINT CHR$ (A(J - N - 1 + N * I));
260 NEXT I
270 PRINT
280 NEXT K
290 PRINT
300 PRINT ,
310 FOR I = 1 TO N
320 PRINT I;
330 NEXT I
340 PRINT
400 PRINT "ENTER COORD"
410 INPUT X
420 IF X ='0 THEN GO TO 500
430 PRINT X
440 LET J = X / 10
450 LET I = X - 10 * J
460 PRINT "ENTER CHAR CODE"
470 INPUT X
480 LET A(J - N - 1 + N * I) = X
490 GO TO 200
500 PRINT "CODES USED:"
510 FOR K = 1 TO N
520 LET J = N + 1 - K
530 PRINT "LINE:"; J,
540 FOR I = 1 TO N
550 PRINT A(J - N - 1 + N * I);" ";
560 NEXT I
570 PRINT
580 NEXT K
```

A new house. A new neighborhood. We've all been through it. Last time, I remembered, it took me two or three years before I met most of the people on my block. This move, I was determined that would not happen.

My solution? An open house a few weeks after I moved in. It forced me to get settled fast; I didn't want my neighbors climbing over packing cartons.

I didn't know how many people to expect and whether I'd be able to entertain them all at once, so I set up two computers (borrowed from work) on the lower floor. One was an Apple, the other a ZX-80.

Naturally the Apple equipped with two ALF music synthesizers playing stereo music (accompanied by spectacular color graphics of the notes being played attracted the most initial attention).

But then some people discovered the Sinclair in the other room. "What's this?" and "It can't be a real computer" were the typical comments.

I had seven or eight games on tapes, some of which I loaded in and demonstrated. Some people were interested. Others started drifting back to the wine and cheese. Then I loaded in *Draw a Picture*.

This was the first program I had ever keyed into the ZX-80. I was looking for a short program to try out the machine, and I was interested in the graphics. This seemed ideal. And so it was.

Within 15 minutes I had made a face. (Why is the first thing one usually drawas with paint, computer or charred stick usually a smiling face?) I changed the expression, ears, nose, eyebrows and all the other features at least five times. What fun I think I skipped lunch that day.

My guests also made faces on the ZX-80. First a happy face. "Oh, let me try" said one previously timid woman. She made a face with a large beard. Another woman felt that a squarish, robot-type face was more appropriate.

At this point I broke through the crowd and quickly "drew" a small dog. After all, I reasoned, computers can do more than draw faces. But alas, they would have none of it, and I was unceremoniously elbowed aside for yet another face.

As the guests drifted out, I noticed that the computer crowd downstairs was showing no such inclination. Interestingly enough, this time the Apple had been deserted and the crowd was around the ZX-80 with *Draw a Picture*.

You too can draw faces with your ZX-80. You can draw small dogs, or even other things like boats, flowers and space ships. But don't try to do it with a crowd standing around; all they want are faces.

Draw a Picture is from *30 Programs for the ZX-80*, published by Melbourne House, Glebe Cottage, Gleebe House, Station Road, Cheddington, Leighton Buzzard, Bedfordshire, England, and costs $14.95 in the U.S. or £ 7.45 in the U.K.

71	72	73	74	75	76	77
61	62	63	64	65	66	67
51	52	53	54	55	56	57
41	42	43	44	45	46	47
31	32	33	34	35	36	37
21	22	23	24	25	26	27
11	12	13	14	15	16	17

When the program says "ENTER COORD," enter a single number from the grid numbering diagram. If you enter 0, the program will give you a summary of what codes you used at each location. When the program says "ENTER CHAR CODE," enter the code of the graphics character you want to appear on the screen. If you want to erase a previous character, enter 0.

AVAILABLE FROM THE KEYBOARD:

2		3	
4		5	
6		7	
8		9	
10		11	

NOT AVAILABLE FROM THE KEYBOARD:

128		129	
130		131	
132		133	
134		135	
136		142	

OTHER USEFUL 'GRAPHIC' SYMBOLS:

6	£	18	-	19	+
20	*	23	)	24	(

Pounds and Pence, Dollars and Cents

Leonard Gaunt

The new ROM for the ZX80 may be available before these words are in print. It remedies a lot of the defects of the original ROM. It gives floating point facilities, trigonometric functions, keyboard access to reversed characters and many other useful functions.

For those who still have the old ROM, there is no way of dealing with pounds and pence, or dollars and cents, is there?

As a matter of fact, the answers are 'yes'.

Leonard Gaunt, 44 Hatherop Rd., Hampton, Middlesex, TW12 2RF.

There are ways of getting round the problems. A useful educational program, for example, is to teach kids to add up dollars and cents, or what have you. That can be done in a relatively simple manner, as Program 1 demonstrates. This is a program that goes on indefinitely setting new sums.

This program is self-explanatory; It asks you to add four randomly produced amounts from 1.01 to 99.99.

It asks you to print the answer using Newline instead of the decimal point.

Subroutines line up the decimal point, produce the 0 after the decimal point where required and add the whole numbers and the decimals separately. Your answer and the correct answer are printed and you are told whether you are right or wrong.

You are then invited to press Newline for another sum or S for you score.

The program produces new sums indefinitely.

```
1 LET J = 0
2 LET K = 0
5 PRINT "ADD"
10 LET T = 0
20 LET T2 = 0
25 FOR N = 1 TO 4
30 LET A = RND(99)
40 LET A2 = RND(99)
50 LET X = A
60 LET Y = A2
70 LET T = T + A
80 LET T2 = T2 + A2
90 GOSUB 2000
100 NEXT N
310 LET X = T
311 LET Y = T2
315 PRINT
320 PRINT
321 PRINT "PRINT YOUR ANSWER"
322 PRINT
323 PRINT "PRESS NEWLINE INSTEAD OF"
324 PRINT
325 PRINT "THE DECIMAL POINT"
330 GOSUB 3000

2000 IF Y < 100 THEN GOTO 2035
2010 LET Y = Y - 100
2020 LET X = X + 1
2030 GOTO 2000
2033 PRINT
2034 PRINT
2035 IF X < 10 AND Y < 10 THEN PRINT , , " ";X;" . 0" ; Y
2040 IF X < 10 AND NOT Y < 10 THEN PRINT , , " ";X;".";  Y
2050 IF NOT X < 10 AND NOT Y < 10 THEN PRINT , , X;".";  Y
2060 IF NOT X < 10 AND Y < 10 THEN PRINT , , X;".0";  Y
2070 RETURN

3000 INPUT A
3005 PRINT , , "-----"
3010 PRINT , , A;".";
3020 INPUT B
3025 IF B < 10 THEN PRINT "0";B
3026 IF NOT B < 10 THEN PRINT B
3031 PRINT
3032 PRINT
3035 PRINT "THE ANSWER IS ";
3040 GOSUB 2000
3070 PRINT
3090 IF A = X AND B = Y THEN GOTO 4000
3100 PRINT "YOU WERE WRONG"
3110 LET K = K + 1
3130 GOTO 4040

4000 PRINT "THAT IS RIGHT"
4010 LET J = J + 1

4040 PRINT
4050 PRINT "PRESS NEWLINE FOR ANOTHER SUM"
4055 PRINT
4056 PRINT "OR S FOR YOUR SCORE"
4060 INPUT Z$
4065 CLS
4070 IF Z$ = "" THEN GOTO 5
4080 IF Z$ = "S" THEN PRINT "YOU GOT  ";J;
     " RIGHT AND "; K;" WRONG"
```

```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT A$
30 CLS
40 LET X = 0
50 PRINT "DO YOU WANT EASY, A LITTLE HARDER, OR DIFFICULT QUESTIONS?"
60 PRINT
70 PRINT "TYPE E FOR EASY; H OR D
80 PRINT
90 INPUT B$
100 CLS
110 PRINT "OK,  ";A$
120 PRINT
130 PRINT "HERE ARE 10 QUESTIONS"
140 FOR Z = 1 TO 10
150 LET A = RND(10)
160 LET B = RND(10)
170 IF A < 3 OR B < 3 THEN GOTO 150
180 IF B$ = "D" THEN LET A = A*2
190 IF B$ = "D" OF B$ = "H" THEN LET B = B*2
200 PRINT
210 PRINT
220 PRINT "WHAT IS ";A;" TIMES ";B
230 INPUT T
240 IF T = A*B THEN GOTO 260
250 GOTO 320
260 CLS
270 PRINT "RIGHT"
280 PRINT
290 LET X = X + 1
300 NEXT Z
310 GOTO 360
320 CLS
330 PRINT "WRONG, ";A$;
335 PRINT ", ";A;" TIMES ";B;" IS ";A*B
340 PRINT
350 NEXT Z
360 PRINT "THAT IS ALL, ";A$
370 PRINT
380 PRINT
390 PRINT "YOU SCORED ";X;" OUT OF 10"
400 PRINT
410 PRINT
420 PRINT "DO YOU WANT ANOTHER 10?"
430 PRINT
440 PRINT
450 PRINT "TYPE Y FOR YES : N FOR NO"
460 INPUT D$
470 IF D$ = "Y" THEN GO TO 30
480 CLS
490 PRINT "THANK YOU, ";A$
500 PRINT
510 PRINT
520 PRINT "GOODBYE FOR NOW"
530 PRINT
540 PRINT
550 PRINT
560 PRINT , "BUT PLEASE COME BACK"
```

Multiplication tables

I haven't yet got round to making the ZX80 multiply and divide decimals but Program 2 is a simple 'multiplication table' teacher. It is a 'talkative' program offering three grades of difficulty, setting ten questions, giving the score and inviting further participation.

*"When you're through Dad, I want to write a program to experiment with the effect of a different coefficient on a plot of a polynomial of the form Y (X(= A*X + B*X + C."*

Basic Accounting With Decimal Currencies 2K

```
1 LET J = 0
2 LET J2 = 0
3 LET H = 0
4 LET H2 = 0
5 LET G = 0
6 LET G2 = 0
7 LET K = 0
8 LET K2 = 0
9 LET L = 0
10 LET L2 = 0
11 LET M = 0
12 LET M2 = 0
15 LET Z = 0
16 LET Z2 = 0
45 PRINT "ACCOUNT     £P    CATEGORY"
47 GOSUB 600
50 INPUT A$
60 IF A$  = "T" OR A$ = "TC" THEN GOTO 270
70 INPUT B
80 INPUT B2
85 IF B < 0 THEN LET B2 = -B2
90 INPUT C
110 PRINT A$;
120 LET I = 15
130 LET X = B
140 LET Y = B2
150 GOSUB 2000
155 GOSUB 1000
157 GOSUB 2050
160 LET Z = Z + B
170 LET Z2 = Z2 + B2
195 LET I = 25
200 PRINT C
210 GOSUB 5000
250 GOTO 50
270 GOSUB 600
290 PRINT "TOTAL";
300 LET I = 15
310 LET X = Z
320 LET Y = Z2
325 GOSUB 9000
330 GOSUB 2000
335 GOSUB 1000
339 GOSUB 2050
340 IF A$ = "TC" THEN GOTO 345
342 GOTO 500
345 PRINT
346 PRINT "CATEGORY TOTALS"
350 LET X=G
360 LET Y=G2
365 GOSUB 8000
368 PRINT "...1",
369 LET X = H
370 LET Y = H2
375 GOSUB 8000
379 PRINT "...2",
380 LET X = J
381 LET Y = J2
385 GOSUB 8000
387 PRINT "...3",
388 LET X = K
389 LET Y = K2
390 GOSUB 8000
395 PRINT "...4",
400 LET X = L
404 LET Y = L2
405 GOSUB 8000
425 PRINT "...5",
430 LET X = M
440 LET Y = M2
450 GOSUB 8000
460 PRINT "...6",
465 GOSUB GOSUB 600
500 INPUT B$
```

Program 3 is very complicated. It keeps accounts, in a way. It prints out the account name or number, adds and subtracts cash amounts, splits them into categories or groups, gives one grand total and six sub-totals. Of course, it is rather limited by the availability of display space. The 'Continue' facility can be used but it really needs a printer.

This program allows the entry of account number, name, etc., plus a column of pounds and pence, or dollars and cents, and a category column.

The amounts can then be totalled in one grand total and six subtotals by category number.

Data entry is in the form:
1 Account number, name, etc. (any string)—newline
2 Pounds/dollars—newline—pence/cents—newline
3 Category of account (any number, normally 1 to 6)—newline

Note that the decimal point is not used but that .50 must be entered as 50 and not 5 as in calculator usage. The entry of a single figure after the decimal point is treated as one-hundredth part, i.e. 5 is treated as 05.

Debits (minus figures) are entered in the normal way e.g. -2566 newline 09. There is no need to put the minus sign before the decimal entry.

After the entry of the category number, the details in 1,2 and 3 are printed. Repeat for as many entries as required.

When all entries have been made, enter:
T newline to obtain main total only OR
TC newline to obtain main total and subtotals

If further entries or corrections are necessary, enter
C newline
and repeat 1,2,3,T or TC. The amounts entered are added to the previous totals.

The program uses several subroutines for tabulation, to persaude the ZX80 to add and subtract decimals and to ensure that the results are printed correctly.

```
505 PRINT
510 IF B$ = "C" THEN GOTO 50
520 STOP
600 FOR N = 1 TO 32
610 PRINT "-";
620 NEXT N
630 RETURN
1000 GOSUB 1040
1010 GOSUB 1050
1020 RETURN
1040 PRINT " " ;
1050 IF I + PEEK(16420) = 33 THEN RETURN
1060 IF I = 0 THEN IF PEEK(16420) < 2 THEN RETURN
1070 GOTO 1040
2000 IF Y < 100 THEN GOTO 2040
2010 LET Y = Y - 100
2020 LET X = X + 1
2030 GOTO 2000
2040 RETURN
2050 IF Y < 10 AND NOT Y < 0 THEN PRINT X;".0";Y;
2055 IF Y < -9 THEN PRINT X;".";Y*-1
2060 IF Y > -10 AND Y < 0 THEN PRINT X ;".0";Y*-1
2065 IF NOT Y < 10 THEN PRINT X;".";Y;
2070 RETURN
5000 IF C = 1 THEN LET G = G + B
5010 IF C = 1 THEN LET G2 = G2 + B2
5020 IF C = 2 THEN LET H = H + B
5030 IF C = 2 THEN LET H2 = H2 + B2
5035 IF C = 3 THEN LET J = J + B
5040 IF C = 3 THEN LET J2 = J2 + B2
5045 IF C = 4 THEN LET K = K + B
5050 IF C = 4 THEN LET K2 = K2 + B2
5055 IF C = 5 THEN LET L = L + B
5060 IF C = 5 THEN LET L2 = L2 + B2
5065 IF C = 6 THEN LET M = M + B
5070 IF C = 6 THEN LET M2 = M2 + B2
5080 RETURN
6000 IF Y > -100 THEN GOTO 6040
6010 LET Y = Y + 100
6020 LET X = X - 1
6030 GOTO 6000
6040 RETURN
6500 LET Y = Y + 100
6510 LET X = X - 1
6520 GOTO 9010
7000 LET X = X + 1
7010 LET Y = Y - 100
7020 GOTO 9020
8000 GOSUB 9000
8100 GOSUB 2000
8020 GOSUB 2050
8030 RETURN
9000 IF Y < 0 AND X < 0 THEN GOSUB 6000
9010 IF X > 0 AND Y < 0 THEN GOTO 6500
9020 IF X < 0 AND Y > 0 THEN GOTO 7000
9030 RETURN
```


The magazine for Sinclair ZX80 users

SYNC

T-SHIRTS!

Proclaim the computer of your choice on your chest with our own Crash Cursor and Sync. Design is white on dark blue shirt for a striking effect. Available in adult S,M,L and XL. $6.00 postpaid in U.S.A.; 3.00 postpaid in U.K. Send U.S. orders to SYNC, 39 East Hanover Ave., Morris Plains, NJ 07950. Send U.K. orders to SYNC, 27 Andrew Close, Stoke Golding, Nuneaton CV13 6EL.

Adding an LED Load Monitor to the ZX80

Cecil Bridges

The input signal level is critical when loading a program from cassette tape. *Save* and *Load* operations can be done with the computer as it exists: One establishes the correct volume control setting on the tape recorder by trial and error. One then secures the volume control in the correct position with sticky tape, or makes a couple of index marks which allow the recorder to be set properly. If one's tape recorder batteries age, or if one tries to load a program recorded by someone else, then a new level has to be re-established by trial and error.

An alternative is to use a load monitor to show when a program to be loaded is at the correct level. Initially, I dedicated an oscilloscipe as a load monitor, but this seemed like overkill, particularly in view of the size discrepancy between the 'scope and the computer. I then added a single LED (light emitting diode) to the tape "ear" input of the computer. The voltage for lighting the LED is just right for loading a program from the cassette tape.

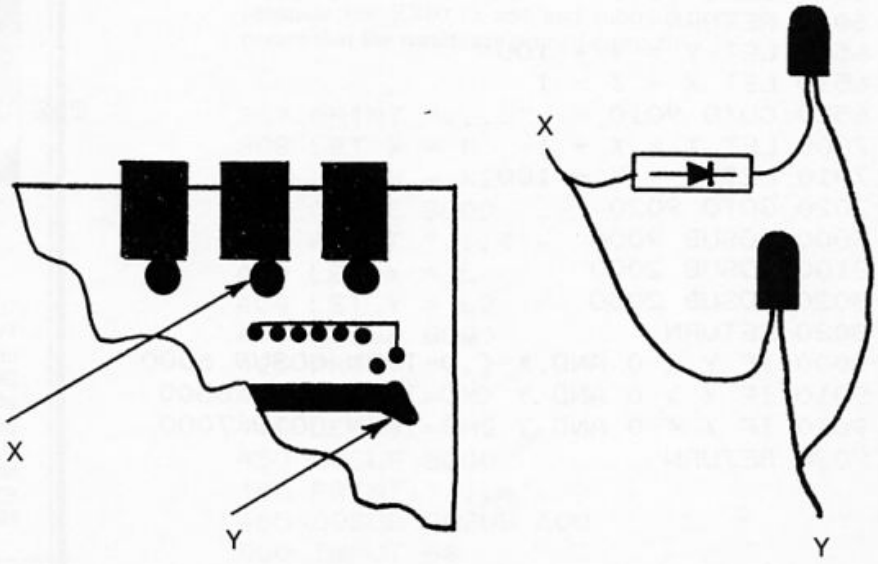
In order to add the LED, it was necessary to get the top off the computer. This was no problem for me because I had already destroyed the plastic rivets in attempting to assemble my Microace, and had substituted small screws. You may want to add the Microace 2K RAM option at a later date, and screws look better than the rivets anyway, so very carefully drill out the five rivets around the outside of the top cover, taking special care not to damage the printed circuit board. Get out your soldering pencil and thin rosin core solder. The LED can be soldered into the printed circuit board from the top if you want to avoid taking out the rivets holding the board to the bottom of the case. The short lead on the LED is attached just below the input ("ear") jack at the point marked X in the illustration and the long lead attaches at the point marked Y. You will probably have to extend the leads of the LED with fine insulated wire. Drill a hole in the top of the case where the LED will stick through. Use insulating tape on the bare lead wires of the LED and assemble the case.

To minimize any interaction with the input circuit, use small LEDs which draw less current and have less capacitance than the large ones. I found mine, red, about 2mm in diameter, at Radio Shack. Be careful when bending the leads of these little LEDs; any strain on them will break their little red plastic cases.

In use, the LED will just light up when the tape recorded program is coming in at the correct level, and will remain dark or barely flicker when it is too low for the computer to accept.

I used the single LED load monitor for a while, but later added two additional components to make a more elegant monitor. A second LED was added in parallel with the first, with a silicon diode in series with it to drop the voltage slightly. See the wiring diagram in the illustration. The second LED then acts as an over-voltage indicator when it is lit. The 3/4 volt forward conduction voltage drop across the silicon diode makes a fairly precise bracket for the input voltage. Solder the second LED and the silicon diode to the lead extensions of the first LED, not to the printed circuit board.

Setting the correct tape recorder volume level for loading is now simple. Given a taped program recorded at some unknown level, the volume level is increased (while the tape recorder is playing back the recorded program) until the first LED lights up and the second remains dark. Rewind the tape and it should then load successfully.



Cecil Bridges, 1248 N. Denver, Tulsa, OK 74106.

New Peripherals from Sinclair

Printer

"Extremely low cost." That's the answer I got when I asked Nigel Searle about the forthcoming Sinclair printer. Relentlessly, I pressed for more.

It is a line-at-a-time printer and only prints a line when the user hits NEWLINE. "Seems awkward," said I. "Extremely low cost," said Nigel.

It has a 32-character line, the same as the display. "That means non-standard paper," said I. "Extremely low cost," said Nigel.

The printer uses a dot matrix and prints the complete ZX80 graphics character set. "Very nice," said I. "Yes," agreed Nigel.

Flat Screen Display

When I interviewed Clive Sinclair last fall, he indicated that the major research program at that point was flat screen television. Nigel Searle informs us that working prototypes have been produced and that a factory is now being built to produce the units. Production units are expected by December 1981.

The CRT measures 3 inches across the diagonal. Sinclair is currently considering a projection device to increase the effective picture size to 5 inches. The CRT tube is an astonishing 3/8" thick.

A standard, but miniature, projection gun shoots the beam parallel to the surface of the screen and it is then bent. The principal itself is not new, however, this is the first time it has been successfully executed.

Clive Sinclair was emphatic that the company would not introduce anything that was not price competitive with existing technology. However, he would not elaborate on the price of the flat screen TV in September when we talked. Now, with four months of additional experience with the unit, Nigel tells us that the original retail target price was $125 or £ 50. This was for a broadcast TV receiver. (Existing conventional 3" units now on the market carry retail prices of $200 and up.) However, more good news. It appears by employing a highly automated production line, the original target price can be beaten down some 40% to $75 or £ 30.

Sinclair Withdraws 8K Basic ROM

We read with interest in *Computer Weekly* that Clive Sinclair has withdrawn the 8K Basic ROM chip for the ZX-80 until the first quarter of 1981. This occurred only a month after it was announced to the world (September 11, 1980). He denies rumors that software bugs are the cause and blames the delay instead on the incorporation of extra functions.

The 8K ROM chip is supposed to directly replace the original 4K ROM even though there is no upward compatability between the two slightly different Basics. When it eventually appears, the 8K ROM will enable the ZX-80 to work in floating point arithmetic to 9-digit accuracy.

The extra functions now being incorporated in the chip mainly consist of a printer driver. The printer is planned for introduction in the first half of 1981.

In explaining the product announcement and subsequent withdrawl, Sinclair stated, "We were caught out. We wanted to bring out the 8K as soon as possible, but on the other hand we did not want to be severely critisized at a later stage when we would have to bring out another version."

8K Basic ROM

Originally announced in September 1980 and then withdrawn, the 8K Basic will probably be out by the time this is printed.

The new one differs from the original in that it incorporates a printer driver routine and one other new feature about which Sinclair is being quite secretive. Some minor functions were cut out of the first version to make way for these two new features.

Sinclair either doesn't know or isn't saying whether the 8K Basic will work in the MicroAce. Apparently there is no love lost between the two companies even though they have a rather close business licensing arrangement. Reputable sources tell us that, indeed, the 8K ROM will work in the MicroAce.

30 Programs for the Sinclair ZX-80 — a Review

David Lubar

30 Programs for the Sinclair ZX-80, by Melbourne House, Glebe Cottage, Glebe House, Station Road, Cheddington, Leighton Buzzard, Bedfordshire, England, distributed in the United States by Image Computer products, 615 Academy Drive, Northbrook, IL 60062, 111 pages, paperback $14.95. 1980.

30 Programs for the Sinclair ZX 80 published by Melbourne House, contains games and utilities that will run in 1K of memory. But the book is more than just a collection of programs; it is a storehouse of valuable techniques and tricks. While many of the programs are entered in the usual way, some are entered in ways that expand the capabilities of the Sinclair. For example, several of the programs come in two parts. The first part accepts inputs which are stored as data for the second part. In "Lunar Lander," these data are used to draw the rocket ship. In "Dr. ZX-80" (see Creative Computing's *More Basic Computer Games* for the original Dr. Z), the data are used to store strings in memory.

These techniques alone make the book worthwhile. Add to this the fun of the games themselves, from a simple "Leap Frog" to the complex programming of "Gomoku," and you have a bargain that should be a part of every Sinclair owner's library.

The listings are presented in clear, large type. The numbers are larger than the letters, thus avoiding confusion between zero and the letter O, as well as one and the letter I. Each program is preceeded by a description which covers any special techniques used. The only shortcoming is the lack of sample runs. It is always nice to see what a program does before going to the touble to enter the whole thing. Also, sample runs can help show where you went wrong if there is an error in your typing. Despite this, the book is highly recommended.

You will find two programs from the book reprinted in this issue of SYNC. "Life" and "Draw a Picture" were reprinted courtesy of Melbourne House Publishers.

Linsac's Game Programs 1 and 2
A Review

Jonathan A. Stein

Linsac of England is now producing software for the Sinclair ZX-80. In addition to publishing the book, *The ZX-80 Companion*, they are marketing a series of educational and game cassettes, all of which are reaching The United States through Image Computer Products, 615 Academy Drive, Northbrook, Illinois 60062. In the U.K. they are available directly from Linsac, 68 Barker Road, Middlebrough TS5 5ES, England.

Although there are five games cassettes packages, we chose to look at the two for the 1K computer, since this is the most common version of the machine. We will reserve reviewing the educational cassettes and the games for the 2K version until a later date. Games Pack One includes *Three Towers*, *Number Guessing*, *Mastermind*, *Sketches*, *Hurkle*, *Nim*, and *Simple Simon*.

Three Towers is a version of the game *Tower of Hanoi*, in which the object is to take six graduated rings and move them from the first of three rods to the third, without ever placing a larger ring on top of a smaller one. The game is difficult until one figures out the system needed to make it work, although it continues to be time consuming. If, however, you make an improper entry, the program is unforgiving, sometimes forcing you to begin again.

The second game on the cassette is *Guess a Number*, which has the player deduce a number between 1 and 100 based on hints offered by the program, which tell the player whether his or her guess is above or below the value of the number trying to be guessed. This game should be particularly interesting to a child, but for an adult it is a bit limited.

Mastermind is a version of the popular board game of the same name. The machine selects a four digit code which you try to solve. After you enter your guess, the computer responds, letting you know how many digits are correct and in the right location, and how many are correct, but improperly situated. The game is extremely challenging, and made more difficult by the way in which it treats duplicated digits. For example: if the code contains two zeros, and you only have one in your guess, the machine will tell you that you have one digit in the right place and one in the wrong place.

In *Sketcher*, an introductory program for computer graphics. The player is given a 31 x 16 grid, on which he can enter designs. As a first encounter with computer drawing, the program is very good, and can be a delight to small children and of interest to adults.

Linsac's version of *Hurkle* is a pleasure to play. The player is given a 15 x 15 grid in which a Hurkle is hiding. To find the Hurkle one must pick X and Y values corresponding to the location of the beast. With only five guesses, you must quickly follow the hints provided by the computer. To keep track of your path, I would recommend writing down both your guesses and the computer hints.

Nim is another game on this cassette, in which the player competes against the computer by taking stones away from several piles. The object is to be the last player to take a stone. The human player has the option to establish the number of piles, and the number of stones in those piles. The game is pleasant, and can often be challenging.

Symbol Simon is the final game in this series. The computer establishes a pattern of letters, numbers, or graphics, depending upon your selection. The game is fairly easy, because it just entails repeating what the program presents.

The second Linsac game package offers several more challenging games and exercises than does the first selection of games. It includes *Nine Lives*, *The Maze Game*, *Plain Sailing*, *Noughts and Crosses*, *Chinese Puzzle*, *Tower of Hanoi*, and *Battleships*.

The first game is *Nine Lives*, a version of the children's game *Hangman*. Two people play, using the machine as both the paper and the scorer. The first player enters a word, and the computer tells the second person the number of letters in that word by displaying a dash for each character. With each incorrect letter entry, one of nine lives is lost. The game continues until the word is completed, or until all of the lives are gone.

The Maze Game, the second game on the tape,is one of the better Linsac games I sampled. The object is for the player to find his way through a constantly changing maze to a treasure. Quite often the maze closes in to make reaching the booty impossible. After the game ends, either by winning the treasure, or by conceding defeat (by entering 0), the computer posts the score out of a possible 18.

Plain Sailing is the third "game" on the cassette. Actually more a mathematical exercise than a game, it entails sailing a dinghy, with a top speed of six knots through a chanel four miles long and two miles wide. The sailor selects the wind speed and direction, as well as the tidal force and direction. After entering this information the player must determine the length of time and number of tacks needed for the journey. The computer then posts the actual number of minutes and tacks required. Although this is really not a game as such, it is extremely interesting, requiring a good deal of thought and attention.

The fourth game, *Noughts and Crosses*, commonly known as *Tick Tac Toe*, offers the player more of a challenge than one usually expects from this game. The ZX-80 version of this old favorite requires a different kind of strategy than when the game is played against a human opponent, because the machine is so consistent. For instance: If your first move is in the center square, the ZX-80 will always play the lower right square. If however, you choose any square but the center, the machine takes the center position. The machine will consistently win unless you take a radical plan to keep you moves on the offensive. Once you are forced into a defensive position, the machine will win, or you will draw.

The next game, *Chinese Puzzle*, is a challenging game requiring a great amount of patience. The object is to slide the alphabet squares around until they are in proper order. Considering there is just one vacant place on the board, this becomes a major task. The game easily becomes frustrating, because it seems that you can never really gain ground, you just switch around the squares which are in the proper positions. The game is fairly good if you are long on patience and endurance.

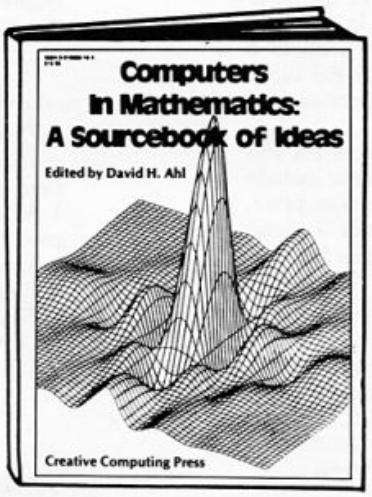
Tower of Hanoi is virtually identical to the *Three Towers* game already reviewed in this article. The primary differences are the way in which the entries are made, and that in this game the game is won by setting up all of the rings on either the second or third rod, rather than just on the third column as in the *Three Towers* game. The game is moderately challenging, but one version is enough.

The final game on the cassette is *Battleships*, which is basically a computerized version of the old board game *Battleship*, in which the object is to sink the enemy ships on a grid marked with X and Y coordinates. The main difference between the ZX-80 version and the board version is that the player does not have a fleet of his own. This game is based totally on trial and error, as one randomly picks X and Y coordinates to fire torpedoes into the enemy waters. The game easily becomes tiresome, because it requires no skill or cunning, just the methodical entering of coordinates.

Both of the Linsac tape packages offer some intersting games and exercises, of which my favorites are: *Hurkle*, *The Maze Game*, *Plain Sailing*, and one of the *Tower* games. Most of the games are made a little eaiser if you use pencil and paper to record your moves. The games packages are definately worth having although several of the games are not terribly challenging.

Sourcebook of Ideas

Many mathematics ideas can be better illustrated with a computer than with a text book.



Consider Baseball cards. If there are 50 cards in a set, how many packs of bubble gum must be purchased to obtain a complete set of players? Many students will guess over 1 million packs yet on average it's only 329.

The formula to solve this problem is not easy. The computer simulation is. Yet you as a teacher probably don't have time to devise programs to illustrate concepts like this.

Between grades 1 and 12 there are 142 mathematical concepts in which the computer can play an important role. Things like arithmetic practice, X-Y coordinates, proving geometic theorems, probability, compounding and computation of pi by inscribed polygons.

Endorsed by NCTM

The National Council of Teachers of Mathematics has strongly endorsed the use of computers in the classroom. Unfortunately most textbooks have not yet responded to this endorsement and do not include programs or computer teaching techniques. You probably don't have the time to develop all these ideas either. What to do?

For the past six years, *Creative Computing* magazine has been running two or three articles per issue written by math teachers. These are classroom proven, tested ideas complete with flowcharts, programs and sample runs.

Teachers have been ordering back issues with those applications for years. However, many of these issues are now sold out or in very short supply.

So we took the most popular 134 articles and applications and reprinted them in a giant 224-page book called *Computers in Mathematics: A Sourcebook of Ideas.*

Ready-to-use-material

This book contains pragmatic, ready to use, classroom tested ideas on everything from simply binary counting to advanced techniques like multiple regression analysis and differential equations.

The book includes many activities that don't require a computer. And if you're considering expanding your computer facilities, you'll find a section on how to select a computer complete with an invaluable microcomputer comparison chart.

Another section presents over 250 problems, puzzles, and programming ideas, more than are found in most "problem collection" books.

Computers in Mathematics: A Sourcebook of Ideas is edited by David Ahl, one of the pioneers in computer education and the founder of *Creative Computing.*

The book is not cheap. It costs $15.95. However if you were to order just half of the back issues from which articles were drawn, they would cost you over $30.

Satisfaction Guaranteed

If you are teaching mathematics in any grade between 1 and 12, we're convinced you'll find this book of tremendous value. If, after receiving it and using it for 30 days you do not agree, you may return it for a full refund plus your return postage.

To order, send your check for $15.95 plus $1.00 postage and handling to Creative Computing Press, Morris Plains, NJ 07950. Visa, MasterCard, and American Express orders may be called in toll-free to 800-631-8112 (in NJ 201-540-0445). School purchase orders should add an additional $1.00 billing fee for a total of $17.95.

Don't put it off. Order this valuable sourcebook today.

creative computing

Morris Plains, NJ 07950
Toll-free **800-631-8112**
(In NJ 201-540-0445)

Resources for the ZX-80 and MicroAce

Presented below is a list of currently available products related to the Sinclair ZX-80. Such a list can never be complete, of course. We intend to run this as an ongoing column and in addition, we intend to expand the entries to include a description of each product. In this and other issues you will find in-depth evaluations and reviews of many of these products.

We welcome entries from manufacturers and readers for the resources column. Please include the name of the item, a brief description, price, and complete data on how to obtain it. Send contributions to SYNC Resources, 39 East Hanover Avenue, Morris Plains, New Jersey 07950.

Computers

Sinclair ZX-80 (Great Britain):
1K assembled — £ 100.00
1K kit without power source — £ 80.00

Science of Cambridge
6 Kings Parade
Cambridge, Cambs CB2 1SN
England

Sinclair ZX-80 (United States):
1K assembled — $200.00

Sinclair Research Limited
50 Staniford Street
Boston, MA 02114

MicroAce Computer:
1K kit — $149.00
2K kit — $169.00

MicroAce
1348 East Edinger
Santa Ana, CA 92075

Hardware

Plug in RAM chips for the ZX-80.
1K — £ 20, 2K — £ 30, 3K — £ 40

Quicksilva
56 Bedford Place
South Hampton, Hants
England

Software and Books

Games and technical information for the Sinclair ZX-80.

Bobker
29 Chedderton Drive
Unsworth, Bury, Lancs
England

Four cassette tapes — £ 3 each:
1. *Moon-landing*, *Hangman*, *Code breaker*, *Intercept*
2. *Bio-rhythm*, *Solitaire*, *Battleship*, *Dice*
3. *Bingo*, *Minefield*, *Remcard*, *Sequences*
4. *Five Mathematics Programs*

Bug-Byte
251 Henley Road
Coventry CV2 1BX
England

1K Games for the ZX-80:
Mastermind, *Hangman*, *Simon*, *Computamath* — £ 2 each, four for £ 5.50 incl.
4K Element Math Package — £ 7.50 incl.

Bush Green Enterprises Limited
80 Dumont Avenue
Point Clear, Essex CO16 8JR
England

ZX-80 Software (games and educational):
Dicethrow, *Mastermind*, *DIV Package*, *Electro Bandit*, *Math Programs*, and *Word Puzzles* — £ 5.00 each, three for £ 12.00

CDS Micro Systems
10 Westfield Close, Tickhill
Doncaster DN 11 9LA
England

Software on cassette for 1K ZX-80:
Simon, *Mind Bender*, *Destroyer*, *Fruit Machine* — £ 5.00, and a general mathematics program for £ 5.00

Econo Soft
4 The Loont
Winsford,Cheshire CW 7 1EU
England

Software for the ZX-80

Tim Hartnell
44-46 Earls Court Road
London W8
England

Software Cont.

ZX-80 game for 1 or 2 players — £ 1.50 in UK

R. Hindle
139 Penrhyn Road
Sheffield, S11 8UP
England

Offers an extensive line of ZX-80 Software. Includes Computer Learning Lab book and tape package — $49.95, a wide selection of game programs, as well as Software produced by Linsac, Melbourne House, and Creative Computing.

Image Computer Products
615 Academy Drive
Northbrook, IL 60062

Games package for 1K ZX-80:
K *Treck*, *Life*, *Mind Master*, *Lunar Lander*, four for $9.95, and an Introductory tape for $2.99

Lamo-Lem
712 Pidgeon Street
San Diego, CA 92114

The ZX-80 Companion — $19.95, and nine programmed cassettes, of both educational material and games — $6.95 each.

Linsac
68 Barker Road
Middlebrough TS5 5ES
England

ZX-80 Active Display — £ 5.00, and software.

K. Macdonald
26 Spiers Close
Knowle, Solihull B93 9ES
England

30 Programs for the ZX-80 — £ 7.45 in the UK, £ 7.95 outside the UK

Melbourne House Publishers
Glebe Cottage, Glebe House
Station Road, Cheddington
Leighton Buzzard, Bedfordshire
England

Software Packages for the ZX-80

The Software House
146 Oxford Street
London W1
England

Games for the ZX-80, three cassette packages — £ 4.95 each

Syntax Software
96 Collingwood Gardens
Giants Hill
Ilford, Essex
England

ZX-80 Magic Book — £ 4.75, and ZX-80 memory and I/O boards

Timedata Limited
57 Swallowdale
Basildon, Essex
England

ZX-80 Teaching Aids:
2X to 10X times tables on cassette — £ 5.00
Higher level teaching programs — £ 5.00

Turner Consultants
Falconsbeck
Holbeck Lane, Windmere, Cumbria
England

20 ZX-80 Programs, a book for £ 5.25 cassette and book for £ 11.00

Zipprint
11 Romsey Road
Winchester, Hants
England

Users Groups and Newsletters

Publishes *Syntax ZX-80*
12 month subscription — $25.00 in The United States, $39.00 in Western Europe, and $30.00 in Canada.

The Harvard Group
RD 2 Box 457
Bolton Road
Harvard, MA 01451

Bi-monthly newsletter, low cost software, subscription — £ 6.00 in UK, £ 10.00 overseas

ZX-80 User's Club
c/o D. Blagden
P.O. Box 159
Kingston Upon Thames, Surrey
KT2 5UQ
England

Seeks to create and share software for the 1K ZX-80. Membership is free (enclose postage). Publishes the newsletter, *Interface*.

National ZX-80 User's Club
44-46 Earls Court Road
London, W8 6EJ
England

SYNC cannot vouch for any of the sources listed above. We welcome objective, in-depth reviews of any of these items and others related to the ZX80.

The Sinclair ZX80 is innovative and powerful. Now there's a magazine to help you get the most out of it.

Get in sync

SYNC magazine is different from other personal computing magazines. Not just different because it is about a unique computer, the Sinclair ZX80 (and kit version, the MicroAce). But different because of the creative and innovative philosophy of the editors.

A Fascinating Computer

The ZX80 doesn't have memory mapped video. Thus the screen goes blank when a key is pressed. To some reviewers this is a disadvantage. To our editors this is a challenge. One suggested that games could be written to take advantage of the screen blanking. For example, how about a game where characters and graphic symbols move around the screen while it is blanked? The object would be to crack the secret code governing the movements. Voila! A new game like Mastermind or Black Box uniquely for the ZX80.

We made some interesting discoveries soon after setting up the machine. For instance, the CHR$ function is not limited to a value between 0 and 255, but cycles repeatedly through the code. CHR$ (9) and CHR$ (265) will produce identical values. In other words, CHR$ operates in a MOD 256 fashion. We found that the "=" sign can be used several times on a single line, allowing the logical evaluation of variables. In the Sinclair, LET X=Y=Z=W is a valid expression.

Or consider the TL$ function which strips a string of its initial character. At first, we wondered what practical value it had. Then someone suggested it would be perfect for removing the dollar sign from numerical inputs.

Breakthroughs? Hardly. But indicative of the hints and kinds you'll find in every issue of **SYNC**. We intend to take the Sinclair to its limits and then push beyond, finding new tricks and tips, new applications, new ways to do what couldn't be done before. **SYNC** functions on many levels, with tutorials for the beginner and concepts that will keep the pros coming back for more. We'll show you how to duplicate commands available in other Basics. And, perhaps, how to do things that can't be done on other machines.

Many computer applications require that data be sorted. But did you realize there are over ten fundamentally different sorting algorithms? Many people settle for a simple bubble sort perhaps because it's described in so many programming manuals or because they've seen it in another program. However, sort routines such as heapsort or Shell-Metzner are over 100 times as fast as a bubble sort and may actually use less memory. Sure, 1K of memory isn't a lot to work with, but it can be stretched much further by using innovative, clever coding. You'll find this type of help in **SYNC**.

Lots of Games and Applications

Applications and software are the meat of **SYNC**. We recognize that along with useful, pragmatic applications, like financial analysis and graphing, you'll want games that are fun and challenging. In the charter issue of **SYNC** you'll find several games. Acey Ducey is a card game in which the dealer (the computer) deals two cards face up. You then have an option to bet depending upon whether you feel the next card dealt will have a value between the first two.

In Hurkle, another game in the charter issue, you have to find a happy little Hurkle who is hiding on a 10 X 10 grid. In response to your guesses, the Hurkle sends our a clue telling you in which direction to look next.

One of the most ancient forms of arithmetical puzzle is called a "boomerang." The oldest recorded example is that set down by Nicomachus in his *Arithmetica* around 100 A.D. You'll find a computer version of this puzzle in **SYNC**.

Hard-Hitting, Objective Evaluations

By selecting the ZX80 or MicroAce as your personal computer you've shown that you are an astute buyer looking for good performance, an innovative design and economical price. However, selecting software will not be easy. That's where **SYNC** comes in. **SYNC** evaluates software packages and other peripherals and doesn't just publish manufacturer descriptions. We put each package through its paces and give you an in-depth, objective report of its strengths and weaknesses.

SYNC is a Creative Computing publication. **Creative Computing** is the number 1 magazine of software and applications with nearly 100,000 circulation. The two most popular computer games books in the world, *Basic Computer Games* and *More Basic Computer Games* (combined sales over 500,000) are published by Creative Computing. Creative Computing Software manufactures over 150 software packages for six different personal computers.

Creative Computing, founded in 1974 by David Ahl, is a well-established firm committed to the future of personal computing. We expect the Sinclair ZX80 to be a highly successful computer and correspondingly, **SYNC** to be a respected and successful magazine.

Order SYNC Today

To order your subscription to SYNC, in the USA send $10 for one year (6 issues), $18 for two years (12 issues) or $24 for three years (18 issues). Send order and payment to the address below or call MasterCard, Visa or American Express orders to our toll-free number.

Subscriptions in the UK are mailed by air and cost £ 10 for one year, £ 18 for two years or £ 25 for three years. Send order and payment to the UK address below.

Canadian and other foreign surface subscriptions cost $15 per year or $27 for two years and should be sent to the USA address.

We guarantee your satisfaction or we will refund your entire subscription price.

Needless to say, we can't fill up all the pages without your help. So send in your programs, articles, hints and tips. Remember, illustrations and screen photos make a piece much more interesting. Send in your reviews of peripherals and software too—but be warned: reviews must be in-depth and objective. We want you to respect what you read on the pages of **SYNC** so be honest and forthright in the material you send us. Of course we pay for contributions—just don't expect to retire on it.

The exploration has begun. Join us.

The magazine for Sinclair ZX80 users

sync

39 East Hanover Avenue
Morris Plains, NJ 07950, USA
Toll free **800-631-8112**
(In NJ 201-540-0445)

27 Andrew Close, Stoke Golding
Nuneaton CV13 6EL, England

SYNC Reader Survey

We at **SYNC** would like to know more about you. That way, we can determine what kinds of material to run in the magazine that will best meet your needs.

In **Sync**, I would like to see:

	very much	okay	forget it
Hardware Tutorials	☐	☐	☐
Software Tutorials	☐	☐	☐
Games	☐	☐	☐
Simulations	☐	☐	☐
Educational Software	☐	☐	☐
Mathematics Software	☐	☐	☐
Graphics Software	☐	☐	☐
Business Software	☐	☐	☐
Device Control Software	☐	☐	☐
Programming Tips	☐	☐	☐
Interfacing Techniques	☐	☐	☐
New Products Information	☐	☐	☐
Hardware Evaluations	☐	☐	☐
Software Evaluations	☐	☐	☐
Fiction	☐	☐	☐
Puzzles and Problems	☐	☐	☐
Advertising	☐	☐	☐

About Software:
☐ I like to mainly write my own
☐ I mainly use software from other sources
☐ A bit of both

Is the ZX 80 or MicroAce your only computer?
☐ Yes
☐ No, I also have a _____

I utilize a computer at (check all that apply)
☐ Work ☐ Home
☐ School ☐ Other

Which of the following peripherals do you intend to purchase for your ZX80 (assuming they become available)?

☐ Additional memory ☐ Graphics Tablet
☐ Floppy Disk ☐ Modem
☐ Small Printer ☐ Music Synthesizer
☐ Letter Quality Printer ☐ Other

What other computer magazines do you regularly read?

_____

_____

Age: ☐ Under 20 ☐ 41-50
☐ 21-30 ☐ 51-60
☐ 31-40 ☐ Over 60

Sex: ☐ Male ☐ Female

What was the main reason you bought a ZX 80 or MicroAce?

_____

_____

Comments: _____

_____

_____

_____

Name _____

Address _____

_____

_____

Return to SYNC, 39 East Hanover Avenue, Morris Plains, N.J. 07950

Are you in SYNC?

If not, you should be. We would like any programs, translations of existing programs, games or tips which you have to pass on to fellow Sinclair ZX-80 or MicroAce owners. Articles are much more lively if accompained by photos (black and white), diagrams, and illustrations. If you do not have an output printer, please type program listings and carefully check them against the listing on the screen. Sample runs should be included with programs rather than just a description of what the program does. Articles should be typed, double space. Your name and address, with phone number should be on first page; all other pages should be numbered. All submissions should include return postage. Payment ranges from $15 to $40 per printed page.

Please send all submissions to:

SYNC
39 E. Hanover Avenue
Morris Plains, New Jersey 07950

Find ZX-80 Owners

Advertise in SYNC, the magazine exclusively dedicated to the Sinclair ZX-80 and The MicroAce. Call or write for details and a rate card. Let SYNC readers know who you are.

SYNC
39 E. Hanover Ave.
Morris Plains, New Jersey 07950

Glossary of Computer Terms

Access Time

The interval of time between the calling for information from a storage address and the delivery of that information. In general, tape has a longer access time than disk.

Address

A label (name or number) that designates a location where information is stored in memory device.

Alphanumeric

A set of symbols. Can be letters (A-Z), and/or special punctuation, mathmatical, or graphic symbols.

Architecture

The internal, preset arrangement or organization of a computer which determines how the computer operates. The interconnections of registers, logic units, control logic, etc. that which makes one microprocessor different from another.

Basic

An acronym for Beginners All Purpose Symbolic Instruction Code. A high-level conversational, interpretive, programming language in wide use. Basic was invented by Kemeny and Kurtz at Dartmouth college in 1963. It permits the use of simple English words and common mathmatical symbols to perform the necessary arithmetic and logical operations to solve problems.

Bit

The smallest amount of information that can exist: equivalent to yes/no, on/off, +/-, or 0/1.

Byte

A collection of bits, usually eight of them.

Code

The relationship between bits and a set of characters. Microcomputers deal only with bits when executing a program. Therefore, letters, numbers, and other human understandable characters must be translated into bits. Each character has a bit code representation.

Coding

Preparing a set of computer instructions.

Command

An instruction given to the system through an input device or peripheral. It is executed as soon as it has been received.

Compatibility

There are two types of compatibility: Program and hardware. Program compatibility refers to the ability to run programs on a variety of computers without change the program language. Hardware compatibility means that various components (printers, disks, keyboards, etc.) may be connected directly without intervening electronic devices.

Console

The operating portion of a unit.

Courseware

A combination of content, instructional design, and the software which causes a computer to implement instructions.

CPS

Cycles Per Second

CPU

Central processing unit. The heart of the computer, controlling what the computer does. It includes three main sections: arithmetic, control, and logic elements. It performs computations and directs functions of the system.

Creative Computing

The second best source of Sinclair ZX-80 information.

Crunch

To make information fit into a smaller place.

CRT

Acronym for Cathode Ray Tube. Similar in appearance to a television screen. Information in the form of characters and graphic designs may be displayed on CRTs at a rapid rate. A CRT terminal usually comes with a keyboard for entering information into the computer.

Cursor

Movable indicator on CRT to indicate a specific character or space that is being displayed. The cursor generally lets the user know where the next character to be typed will appear.

Data

The information given to or received from a computer.

Debug

Process of finding, locating, and correcting mistakes or errors in a program that might create problems or provide inaccurate information.

Digit

Either a zero or a one in the binary number system.

Direct Memory Access (DMA)

A technique for rapidly moving data from the microprocessor to a storage device such as a disk. DMA is a accomplished at the direction of a program. Not all microcomputers permit DMA.

Document

A written description of a piece of software or hardware. It can also be used as a verb which is the process of producing such a description.

Driver

Small program which controls peripheral devices and their interface with the CPU.

Editor

A program which allows changing, modification, or movement of programming statements. It allows the programmer to write and modify instructions using the microprocessor and a terminal as a very sophisticated typewriter.

Error

Difference in value between actual response and desired response in the performance of a controlled machine, system, or process.

Execute

The running of a computer program.

External Storage

Auxiliary storage such as tape or disk.

File

Collection of related data.

File Name

Number/letter of characters that identify a file.

Flow Charting

A programming technique of using shaped blocks to indicate the sequence of operations in a program.

Frequency

Rate at which anything recurs.

GIGO

Garbage in Garbage Out. Implies that misinformation applied to the CPU will result in misinformation output.

Glitch

An intermittent bug

Graphics

Characters that can be used to form figures, shapes, and forms on the CRT or printer. In addition to letters and numbers, a computer may have a graphic set, so arrange that they can be combined to form almost any desired figure.

Hard Copy

Data or information printed on paper. Used to distinguish between printed information and the temporary image found on the CRT.

Hardware

Mechanical, magnetic, electrical, and electronic devices which make up a computer. The physical equipment that goes into a computer system, consisting of the central processing unit plus all peripherals.

IC

Integrated Circuit. A plastic or ceramic body five cm long, two cm wide, and three mm thick, with up to 40 leads extending from it. Inside the body is a chip. The body protects the chip, and the leads allow electrical connection of the chip to other components. The word "chip" is not used to refer to the entire IC.

Input

Information going into the computer or into a peripheral. The same data may be output from one part of the computer and input to some other part of the computer. When using this word, specify what the data are input to or output from.

Instruction

A set of bits, or a command, which will cause a computer to perform certain prescribed operations.

Interactive

System capable of two-way communications with a user during operation. A system is interactive if it responds to the user quickly—usually less than a second. All personal computer systems are interactive.

Interface

An electronic circuit used to connect one electrical device to another electrical or mechanical device to allow the flow of data between units. It refers to the matching or interconnecting of systems or devices having different functions.

Internal Storage

Memory system which is part of the computer, as opposed to external tape or disk storage.

Interpreter

A program used to translate languages at the time of processing.

I/O

Input/output of information in a computer system. Examples of I/O devices are a keyboard, a floppy disk drive, and a printer.

Language

A format by which a programmer can communicate more efficiently with a computer where predetermined commands will yield requested actions. Basic is one of the most popular languages.

Low Resolution

Graphics composed of coarse blocks.

Macro-instruction

An instruction which causes the computer to execute one or more other instructions. These "other instructions" are called micro-instructions.

Main Memory

That memory which is directly accessible to the computer. It contains the operating system, programs, and data being processed. In a microcomputer, main memory is referred to as RAM or ROM.

Memory

The integrated circuits of a computer which store information in a microcomputer, these are referred to as RAM and ROM.

Memory Chip

A chip which stores data in the form of electrical charges.

Microprocessor

An integrated circuit that can execute instructions. It is one component of a microcomputer. It is the brains of the central processing unit (CPU).

Microcomputer

A hardware configuration usually acquired in one of three ways: 1) by constructing several components from individual electronic parts (as in building a stereo system from a kit); 2) by connecting several already constructed components (as in purchasing a separate amplifier, speaker and turntable); or 3) by purchasing a unit with built in components (as in buying a complete stereo system in one package, plugging it in, and using it immediately). The end-product of the microcomputer is information. It records this information, puts it into meaningful terms, communicates it, stores it, and retrieves it when needed. It usually includes the microprocessing unit, a keyboard for entering data, and a cassette tape recorder or a disk for storing programs, and a T.V.-like screen for displaying results.

Modulator

An electronic device that allows a normal television set to be used as the video display unit (CPU).

Monitor

1) A video display unit which uses a cathode ray tube to generate characters. It looks much like a normal TV set; however, the monitor has a much higher degree of resolution, which permits a clear formation of very small characters on the screen.

2) A program which oversees the operation of other programs.

Noise

Refers to inaccurate data transmission. This causes typographical errors in output.

Output

Information emanating from a display unit such as a CRT or printer.

Peripheral Device

A device, such as a printer, mass storage unit, or keyboard, which is an accessory to a microprocessor and which transfers information to and from the microprocessor.

Personal Computer

A microcomputer designed for use by an individual for entertainment, instruction, and book keeping chores.

Printer

A peripheral device which accepts output data from the microprocessor and prints characters on paper. Printers are defined as impact or non-impact depending on the means by which a character is formed on the paper. Impact printers strike the paper through a ribbon in a manner similar to a typewriter. Non-impact printers form characters by various means such as heat, electrical charges, or spraying ink.

Program

A series of instructions to a computer which cause the computer to solve a problem or perform a task.

RAM

An acronym for Random Access Memory. Any memory which can be written on or read from by a program in which the memory locations can be accessed in random sequence. RAM can be erased and reprogrammed by the programmer as frequently as necessary. RAM size is expressed as a quantity of bytes, such as 4K (4,000 bytes). RAM may be expaned by adding memory chips or memory boards.

Register

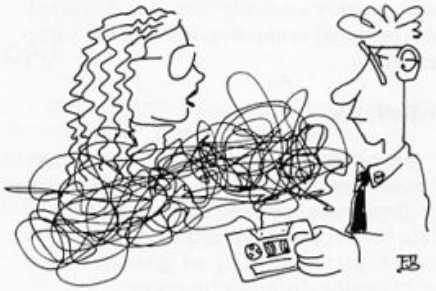
A temporary storage device located in the microprocessor which can hold computer bits or words.

Response Time

The interval of time required for the microprocessor to respond to an instruction or an input from a peripheral such as the keyboard. In an educational environment, the time interval from the activation of the keyboard to a display on the CRT should be less than three seconds.

RF Modulator

See Modulator



"Back up..what back up?"

Run

Jargon for execute, (There is a run key on the ZX-80).

Software

Refers to programs and accompanying documentation. Software is stored on tape cassettes or disks when not being used by the computer. The computer reads the software into its memory in order to use the programs.

Storage Capacity

The quantity of bytes a storage device can hold. It is usually expressed in kilobytes (1024 bytes), which is abbreviated K. Thus, a disk is said to have a storage capacity of 400K (400,000 bytes). This can be understood as 400,000 characters or letters, numbers, spaces, etc.

Storage device

A peripheral device which holds information. This includes tapes and disks.

Store

This term refers to the process of placing data onto some type of storage device. Usually the data are to be kept permanently; therefore, they are placed in a non-volatile memory such as a tape, disk, or static memory.

SYNC

To Sinclair ZX-80 owners as the Koran is to Moslems.

Terminal

A peripheral device which facilitates human communications with a computer. Usually it consist of a keyboard with alphabetic and numeric characters coupled with a printing mechanism or a CRT. One enters information via the keyboard: the computer responds via the printer or CRT.

Video Display Unit

A component of a microcomputer system which displays the output on a screen similar to a TV screen. A television monitor is a type of video display unit.

Many of the definitions are courtesy of Shirley Douglas and Gary Neights, The Pennsylvania Department of Education 33 Market Street, Harrisburg, PA 17126.

A REMARKABLE MAGAZINE

David Ahl, Founder and
Publisher of Creative Computing

creative computing

"The beat covered by Creative Computing is one of the most important, explosive and fast-changing."—Alvin Toffler

You might think the term "creative computing" is a contradiction. How can something as precise and logical as electronic computing possibly be creative? We think it can be. Consider the way computers are being used to create special effects in movies—image generation, coloring and computer-driven cameras and props. Or an electronic "sketchpad" for your home computer that adds animation, coloring and shading at your direction. How about a computer simulation of an invasion of killer bees with you trying to find a way of keeping them under control?

Beyond Our Dreams

Computers are not creative per se. But the way in which they are used can be highly creative and imaginative. Five years ago when *Creative Computing* magazine first billed itself as "The number 1 magazine of computer applications and software," we had no idea how far that idea would take us. Today, these applications are becoming so broad, so all-encompassing that the computer field will soon include virtually everything!

In light of this generality, we take "application" to mean whatever can be done with computers, *ought* to be done with computers or *might* be done with computers. That is the meat of *Creative Computing*.

Alvin Toffler, author of *Future Shock* and *The Third Wave* says, "I read *Creative Computing* not only for information about how to make the most of my own equipment but to keep an eye on how the whole field is emerging.

Creative Computing, the company as well as the magazine, is uniquely lighthearted but also seriously interested in all aspects of computing. Ours is the magazine of software, graphics, games and simulations for beginners and relaxing professionals. We try to present the new and important ideas of the field in a way that a 14-year old or a Cobol programmer can understand them. Things like text editing, social simulations, control of household devices, animation and graphics, and communications networks.

Understandable Yet Challenging

As the premier magazine for beginners, it is our solemn responsibility to make what we publish comprehensible to the newcomer. That does not mean easy; our readers like to be challenged. It means providing the reader who has no preparation with every possible means to seize the subject matter and make it his own.

However, we don't want the experts in our audience to be bored. So we try to publish articles of interest to beginners and experts at the same time. Ideally, we would like every piece to have instructional or informative content—and some depth—even when communicated humorously or playfully. Thus, our favorite kind of piece is acessible to the beginner, theoretically non-trivial, interesting on more than one level, and perhaps even humorous.

David Gerrold of *Star Trek* fame says, "*Creative Computing* with its unpretentious, down-to-earth lucidity encourages the computer user to have fun. *Creative Computing* makes it possible for me to learn basic programming skills and use the computer better than any other source.

Hard-hitting Evaluations

At *Creative Computing* we obtain new computer systems, peripherals, and software as soon as they are announced. We put them through their paces in our Software Development Center and also in the environment for which they are intended—home, business, laboratory, or school.

Our evaluations are unbiased and accurate. We compared word processing printers and found two losers among highly promoted makes. Conversely, we found one computer had far more than its advertised capability. Of 16 educational packages, only seven offered solid learning value.

When we say unbiased reviews we mean it. More than once, our honesty has cost us an advertiser—temporarily. But we feel that our first obligation is to our readers and that editorial excellence and integrity are our highest goals.

Karl Zinn at the University of Michigan feels we are meeting these goals when he writes. "*Creative Computing* consistently provides value in articles, product reviews and systems comparisons . . . in a magazine that is fun to read."

Order Today

To order your subscription to *Creative Computing* send payment to the appropriate address below. Customers in the continental U.S. may call toll-free to charge a subscription to Visa, MasterCard or American Express.

Term	*USA*	*Canada and Foreign Surface*	*Foreign Air*
1 year	$20	$29 or £12.50	$50 or £21
2 years	$37	$55 or £24.00	$97 or £41
3 years	$53	$80 or £34.50	$143 or £61

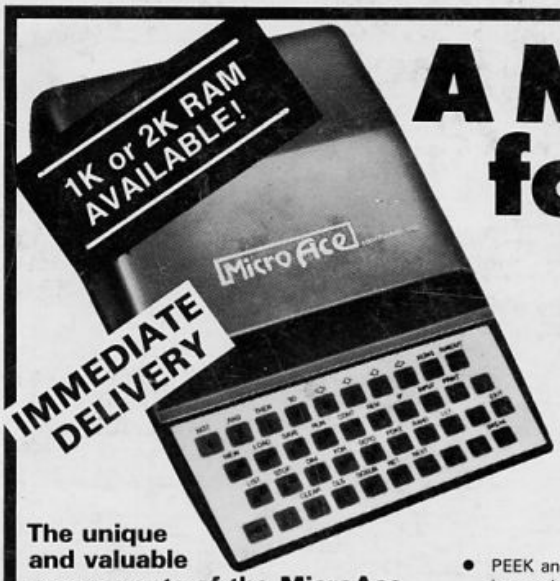
We guarantee your satisfaction or we will refund your entire subscription price.

Join over 80,000 subscribers like Ann Lewin, Director of the Capital Children's Museum who says, "I am very much impressed with *Creative Computing*. It is helping to demystify the computer. Its articles are helpful, humorous and humane. The world needs *Creative Computing*."

creative computing

P.O. Box 789-M
Morristown, NJ 07960
Toll-free **800-631-8112**
(In NJ 201-540-0445)

27 Andrew Close, Stoke Golding
Nuneaton CV13 6EL, England



Licenced by Sinclair Research Ltd.

A Microcomputer for everyone at a Micro Price

The MicroAce - a new generation of miniature computers

A COMPLETE COMPUTER for $149.00 for 1K Kit

Post and Packing FREE

(Add 6% Tax for Shipments inside California)

The unique and valuable components of the MicroAce

The MicroAce is not just another personal computer. Quite apart from its exceptionally low price, the MicroAce has two uniquely advanced components: the powerful BASIC interpreter, and the simple teach yourself BASIC manual.

The unique versatile BASIC interpreter offers remarkable programming advantages:

- **Unique 'one-touch' key word entry: the MicroAce eliminates a great deal of tiresome typing. Key words (RUN, PRINT, LIST, etc.) have their own single-key entry.**
- Unique syntax check. Only lines with correct syntax are accepted into programs. A cursor identifies errors immediately. This prevents entry of long and complicated programs with faults only discovered when you try to run them.
- Excellent string-handling capability — takes up to 26 string variables of any length. All strings can undergo all relational tests (e.g. comparison). The MicroAce also has string input — to request a line of text when necessary. Strings do not need to be dimensioned.
- Up to 26 single dimension arrays.
- FOR/NEXT loops nested up 26.
- Variable names of any length.
- BASIC language also handles full Boolean arithmetic, conditional expressions, etc.
- Exceptionally powerful edit facilities, allows modification of existing program lines.
- Randomise function, useful for games and secret codes, as well as more serious applications
- Timer under program control.
- PEEK and POKE enable entry of machine code instructions, USR causes jump to a user's machine language sub-routine.
- High-resolution graphics with 22 standard graphic symbols.
- All characters printable in reverse under program control.
- Lines of unlimited length.

'Excellent value' indeed!

For just $149.00 (excluding handling charge) you get everything you need to build a personal computer at home... PCB, with IC sockets for all ICs; case; leads for direct connection to a cassette recorder and television (black and white or color); everything!

Yet the MicroAce really is a complete, powerful, full-facility computer, matching or surpassing other personal computers at several times the price.

The MicroAce is programmed in BASIC, and you can use it to do quite literally anything, from playing chess to managing a business.

The MicroAce is pleasantly straightforward to assemble, using a fine-tipped soldering iron. It immediately proves what a good job you've done: connect it to your TV ... link it to the mains adaptor ... and you're ready to go.

Fewer chips, compact design, volume production-more power per Dollar!

The MicroAce owes its remarkable low price to its remarkable design: the whole system is packed on to fewer, newer, more powerful and advanced LSI chips. A single SUPER ROM, for instance, contains the BASIC interpreter, the character set, operating system, and monitor. And the MicroAce 1K byte RAM (expandable to 2K on board) is roughly equivalent to 4K bytes in a conventional computer — typically storing 100 lines of BASIC. (Key words occupy only a single byte.)

The display shows 32 characters by 24 lines.

And Benchmark tests show that the MicroAce is faster than all other personal computers.

No other personal computer offers this unique combination of high capability and low price.

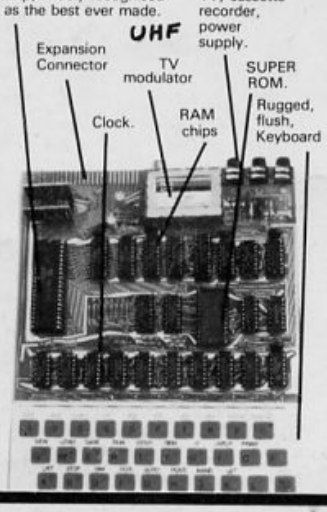
The MicroAce teach-yourself BASIC manual.

If the features of the BASIC interpreter mean little to you-don't worry. They're all explained in the specially-written book *free* with every kit! The book makes learning easy, exciting and enjoyable, and represents a complete course in BASIC programming-from first principles to complex programs. (Available separately-purchase price refunded if you buy a MicroAce later.)

A hardware manual is also included with every kit.

The MicroAce Kit: $149.00 with IK COMPLETE $169.00 with 2K

Demand for the MicroAce is very high: use the coupon to order today for the earliest possible delivery. All orders will be despatched in strict rotation. If you are unsuccessful in constructing your kit, we will repair it for a fee of $20.00, post and packing FREE. Of course, you may return your MicroAce as received within 14 days for a full refund. We want you to be satisfied beyond all doubt — and we have no doubt that you will be.



Your MicroAce kit contains...

- Printed circuit board, with IC sockets for all ICs.
- Complete components set, including all ICs-all manufactured by selected world-leading suppliers.
- New rugged keyboard, touch-sensitive, wipe-clean.
- Ready-moulded case.
- Leads and plugs for connection to domestic TV and cassette recorder. (Programs can be SAVEd and LOADed on to a portable cassette recorder.)
- Mains adaptor of 600 mA at 9VDC nominal unregulated.
- FREE course in BASIC programming and user manual.

JOIN THE REVOLUTION - DON'T GET LEFT BEHIND - ORDER YOUR MICRO ACE NOW!!

Send Check, Money Order or quote your Credit Card No. to:
MicroAce 1348 East Edinger, Santa Ana, California, Zip Code 92705.
or phone (714) 547 2526 quoting your Credit Card Number.

F

Quantity	Description	Unit Price	TOTAL
	MicroAce Kit 1K	$149.00	
	MicroAce Kit 2K	$169.00	
	Manual	$10.00	
	1K Upgrade Kit	$29.00	
Shipments inside California add 6% TAX		**TOTAL**	

☐ Amex.
☐ Diners
☐ Check
☐ Money Order
☐ Master Charge
☐ Visa

Card No. _____

_____

Exp. Date _____

Name _____

Address _____

City _____ State _____ Zip _____