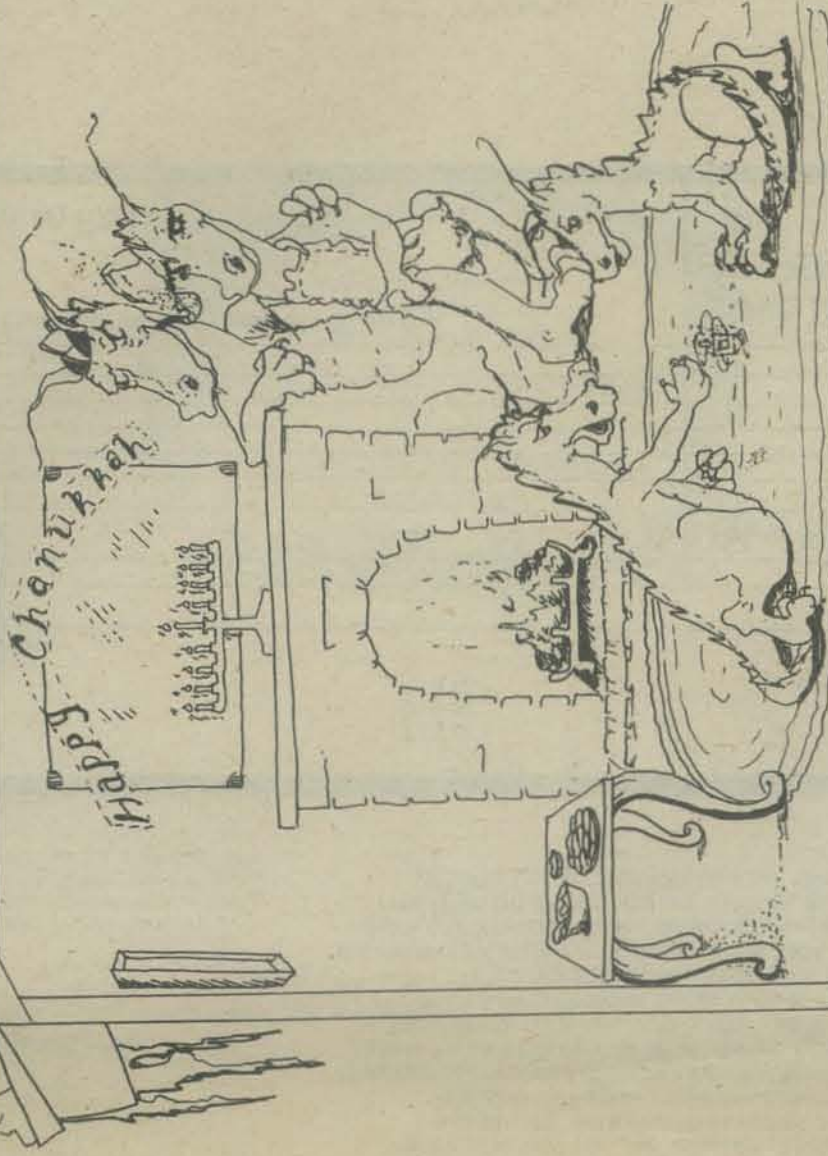


Vol. 4 No. 3  
November 1975

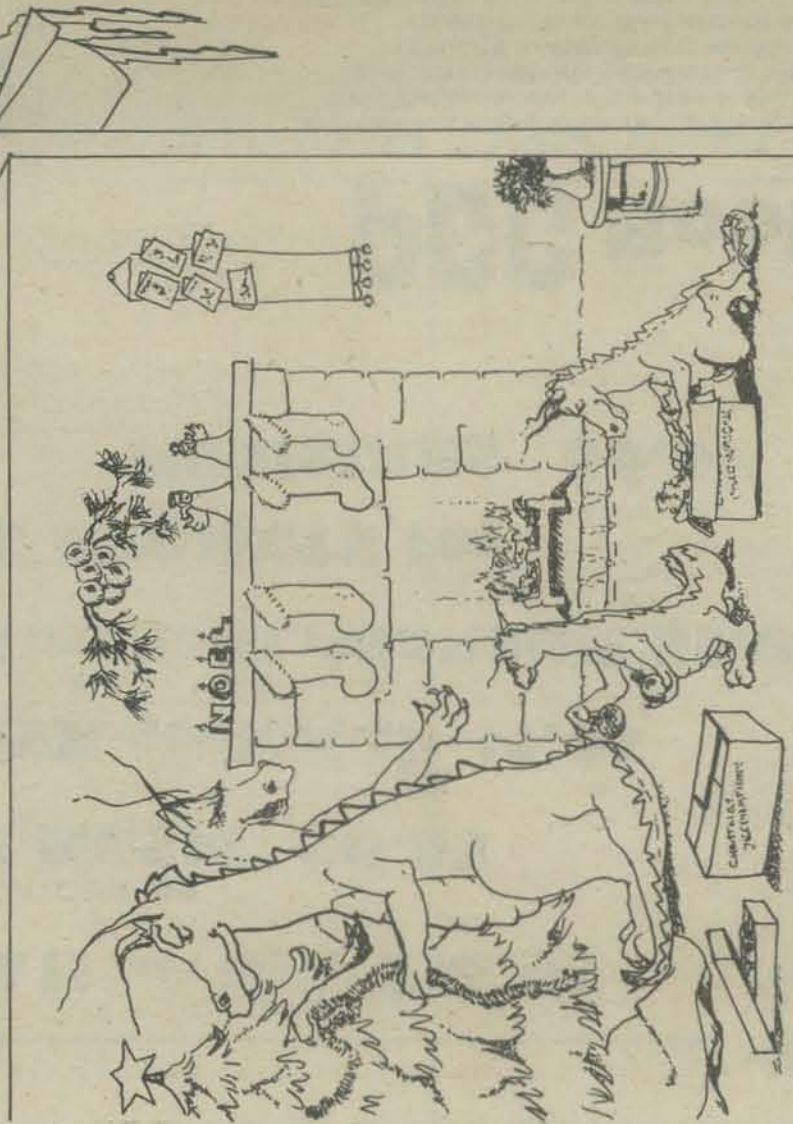
People's Computer Company



Menlo Park, Ca. 94025

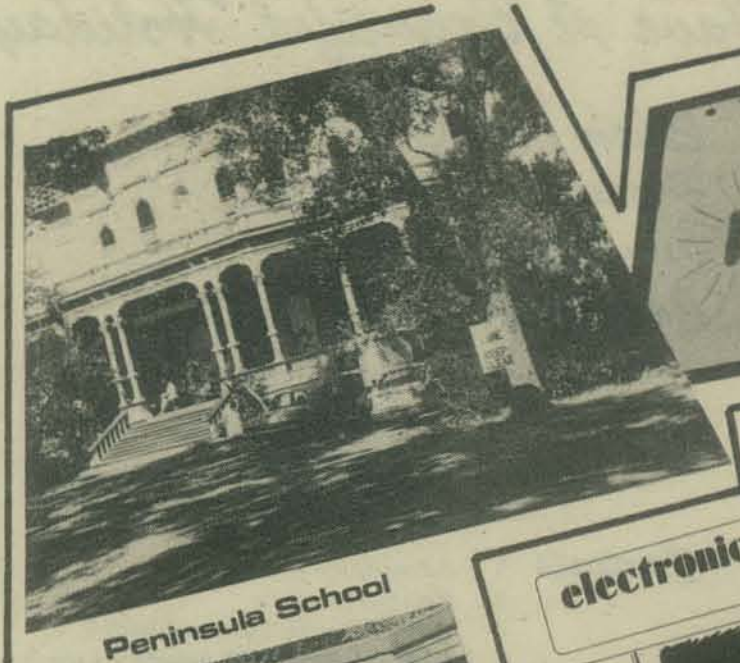
PO Box 310

PCC



Non-Profit Org.  
U.S. POSTAGE  
PAID  
Permit No. 427  
Menlo Park, CA

Robert Zeidman  
9801 Clark St.  
Phila, Pa. 19115

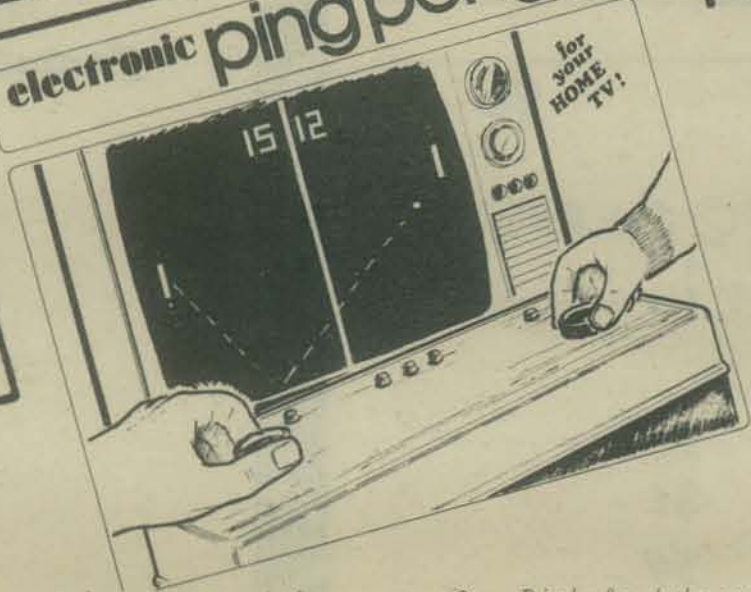


Peninsula School



# electronic ping pong!

for your HOME TV!



## KIDS BUILDING KITS

by Bob Albrecht



One day, not long ago, Dan Pichulo (whose picture you can probably find somewhere on these two pages) wandered into PCC. In one hand he carried a small TV, in the other a small rectangular parallelepiped with switches and knobs. We gathered around curiously as he connected the box to the TV. Dan flipped a switch. On the screen appeared the ingredients of a computerized ping-pong game! We played . . . and after playing, asked him what was in the box. As you might expect, inside was a PC board (Printed Circuit board) with lots of electronic nick-nacks attached to it with strange names like resistors, diodes, capacitors, integrated circuits and such.

And thus begins our first story on Kids Building Kits. We talked Dan out of two kits and some of his time, then asked Peninsula School for a few kit-building hours with 5th, 6th, 7th and 8th grade students. On these pages, you see the beginning of the project, people attaching electronic components to the PC board and soldering them into place. After 6 hours spread over 2 weeks we almost, but not quite, finished. We'll tell you what happened next issue.

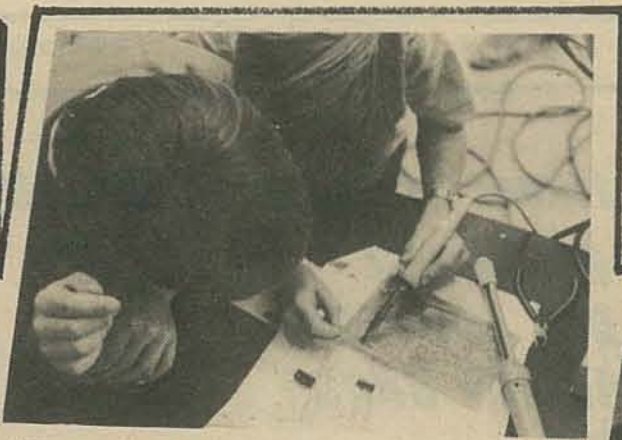
If you know about similar projects, please tell us about them!

For information on the Ping Pong Kit, write to

Dan Pichulo  
Visulex  
P.O. Box 4204  
Mountain View, CA. 94040

# 2





Peninsula School was founded in September, 1925, by a small group of parents who saw the need for a school which would "meet the needs of the individual child and would test the validity of new principles of education by progressive experiment." The involvement of parents in the educational process as well as in the business operation was a primary concern of those founding parents. Now, as much as in those early days, parents help provide the energy which keeps Peninsula School unique.

Peninsula School functions as a cooperative in the sense that parents and staff work together toward the common goal of educating our children. A large proportion of the teaching, administrative, and maintenance work is performed by parents. Many positions are salaried, such as the Director, Assistant Treasurer, classroom teachers, and some special teachers. Other positions, called participating scholarship jobs, are filled by a parent who works in exchange for tuition. These include teaching assistants, some specialty teachers, office work, and maintenance work. The rest, also an integral part of our children's education and of the functioning of the school, is done by volunteers:

- \*driving and helping on day class trips and overnights
- \*volunteering in classrooms, regularly or for special projects
- \*teaching special skills, such as languages, dance ...
- \*organizing and working on events in cooperation with Parents Events
- \*helping with administrative, publicity, or office work
- \*helping maintain grounds and buildings
- \*helping with fund raising

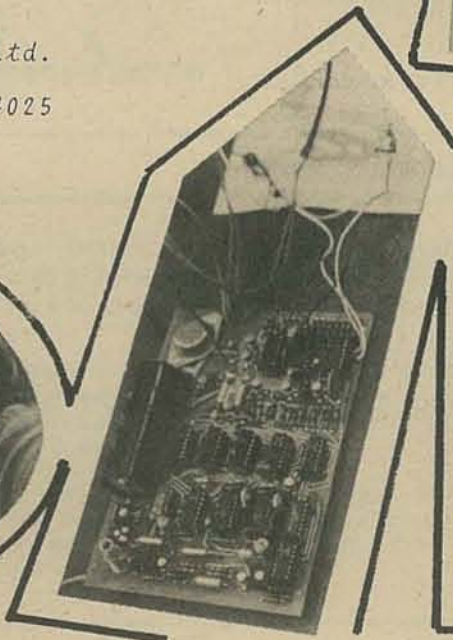
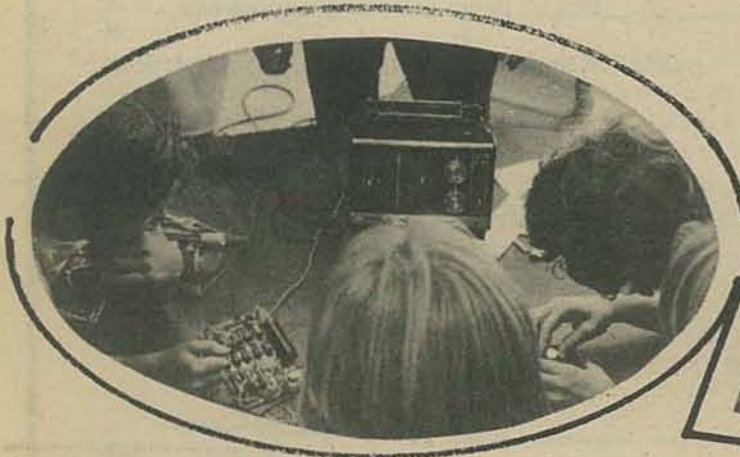
For information, write to:

Peninsula School, Ltd.  
 Peninsula Way  
 Menlo Park, CA. 94025

3



Dan Pichulo



# NUMBER

OK, I HAVE A NUMBER. START GUESSING! WHAT IS YOUR GUESS? 10!



TOO SMALL TRY A LARGER NUMBER! WHAT IS YOUR GUESS? 80!



TOO BIG! TRY A SMALLER NUMBER. WHAT IS YOUR GUESS? 65!



TOO SMALL! TRY A LARGER NUMBER. WHAT IS YOUR GUESS? 68!



TOO BIG! TRY A SMALLER NUMBER! WHAT IS YOUR GUESS? 67!



YOU GUESSED IT! IN ONLY FIVE GUESSES!



ARE YOU REALLY SMART? OR JUST LUCKY?



## What flavor BASIC do you speak?

- HP 2000?
- DEC EDUSYSTEM?
- ALTAIR 8800?

There are many different BASICs. So, we will do a series comparing the three mentioned above plus any stuff that you people out there tell us about your BASIC.

We start with the simple game of NUMBER and write several programs to play the game. Our first program is in "standard BASIC." It will work on

- HP 2000 BASIC
- DEC EDUSYSTEM BASIC (but add 105 RANDOM)
- ALTAIR 8800 BASIC

```

100 REM *** NUMBER - A NUMBER GUESSING GAME (NUM01)
110 LET X=INT(100*RND(1))+1
120 PRINT
130 PRINT "I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
150 PRINT
160 PRINT "YOUR GUESS";
170 INPUT G
180 IF G<X THEN 210
190 IF G>X THEN 230
200 IF G=X THEN 250
210 PRINT "TOO SMALL. TRY A BIGGER NUMBER."
220 GO TO 150
230 PRINT "TOO BIG. TRY A SMALLER NUMBER."
240 GO TO 150
250 PRINT "YOU GUESSED IT!!! LET'S PLAY AGAIN."
260 GO TO 110
999 END

```

← BELLS here \*

← BELLS here \*

RUN

I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100.  
GUESS MY NUMBER!!!

YOUR GUESS? 50  
TOO SMALL. TRY A BIGGER NUMBER.

YOUR GUESS? 75  
TOO BIG. TRY A SMALLER NUMBER.

YOUR GUESS? 62  
TOO SMALL. TRY A BIGGER NUMBER.

YOUR GUESS? 68  
TOO SMALL. TRY A BIGGER NUMBER.

YOUR GUESS? 72  
TOO SMALL. TRY A BIGGER NUMBER.

YOUR GUESS? 73  
YOU GUESSED IT!!! LET'S PLAY AGAIN.

I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100.  
GUESS MY NUMBER!!!

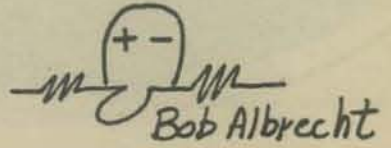
YOUR GUESS?

```

*****
*
* To ring the BELL on a TTY press
*
* (CTRL) and (BELL)
*
*
* together. As you see, nothing is printed,
* but the BELLS do ring. We like lots of
* BELLS in our game-playing programs.
*
*****

```

We will do stuff like this every issue, so send us some ideas!

 Bob Albrecht

**And now...a programming trick!**

For EDUSYSTEM BASIC, add  
105 RANDOM

```
100 REM *** NUMBER - A NUMBER GUESSING GAME (NUM02)
110 LET X=INT(100*RND(1))+1
120 PRINT
130 PRINT "I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
150 PRINT
160 PRINT "YOUR GUESS":
170 INPUT G

180 ON SGN(G-X)+2 GO TO 210,250,230

210 PRINT "TOO SMALL. TRY A BIGGER NUMBER."
220 GO TO 150
230 PRINT "TOO BIG. TRY A SMALLER NUMBER."
240 GO TO 150
250 PRINT "YOU GUESSED IT!!! LET'S PLAY AGAIN."
260 GO TO 110
999 END
```

This one line does the work of these three lines

```
180 IF G<X THEN 210
190 IF G>X THEN 230
200 IF G=X THEN 250
```

This works for EDUSYSTEM BASIC and ALTAIR BASIC.

For HP 2000 BASIC, change Line 180 to  $\rightarrow$  180 GO TO SGN(G-X)+2 OF 210,250,230

OK. For you people with the puzzled expressions, here is how it works.

\* If G is less than X, then G - X is negative. In this case, SGN(G - X) is -1 and SGN(G - X) + 2 is 1. So, the ON ... GO TO ... sends the computer to Line 210.

\* If G is equal to X, then G - X is zero. In this case, SGN(G - X) is 0 and SGN(G - X) + 2 is 2. So, the ON ... GO TO ... sends the computer to Line 250.

\* If G is greater than X, then G - X is positive. In this case, SGN(G - X) is 1 and SGN(G - X) + 2 is 3. So, the ON ... GO TO ... sends the computer to Line 230.

**180 ON SGN(G-X)+2 GO TO 210, 250, 230**



**Next?** Multiple statements per line. Here are crunched programs for EDUSYSTEM BASIC and ALTAIR BASIC.

EDUSYSTEM BASIC uses \ between statements (SHIFT L)

```
100 REM *** NUMBER - A NUMBER GUESSING GAME (NUM03)
105 RANDOM
110 LET X=INT(100*RND(1))+1 \ PRINT
130 PRINT "I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
150 PRINT \ PRINT "YOUR GUESS": \ INPUT G
180 ON SGN(G-X)+2 GO TO 210,250,230
210 PRINT "TOO SMALL. TRY A BIGGER NUMBER." \ GO TO 150
230 PRINT "TOO BIG. TRY A SMALLER NUMBER." \ GO TO 150
250 PRINT "YOU GUESSED IT!!! LET'S PLAY AGAIN." \ GO TO 110
999 END
```

\ separates statements

ALTAIR BASIC uses : between statements.

```
100 REM *** NUMBER - A NUMBER GUESSING GAME (NUM03)
110 LET X=INT(100*RND(1))+1 : PRINT
130 PRINT "I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
150 PRINT : PRINT "YOUR GUESS": : INPUT G
180 ON SGN(G-X)+2 GO TO 210,250,230
210 PRINT "TOO SMALL. TRY A BIGGER NUMBER." : GO TO 150
230 PRINT "TOO BIG. TRY A SMALLER NUMBER." : GO TO 150
250 PRINT "YOU GUESSED IT!!! LET'S PLAY AGAIN." : GO TO 110
999 END
```

: separates statements

Preview of coming attraction! Next issue we will talk about strings in INPUT statements and statements following THEN in IF-THEN statements. Here is a look-ahead for you ALTAIR fans.

```
100 REM *** NUMBER - A NUMBER GUESSING GAME (NUM04)
110 LET X=INT(100*RND(1))+1 : PRINT
130 PRINT "I'M THINKING OF A WHOLE NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
150 PRINT : INPUT "YOUR GUESS":G
180 IF G<X THEN PRINT "TOO SMALL. TRY A BIGGER NUMBER.":GOTO 150
190 IF G>X THEN PRINT "TOO BIG. TRY A SMALLER NUMBER.":GOTO 150
250 PRINT "YOU GUESSED IT!!! LET'S PLAY AGAIN.":GOTO 110
999 END
```

# NUMBER in ASSEMBLY LANGUAGE for the 8080

by

BERNARD GREENING

```

;
;
; NUMBERS
;
; THIS PROGRAM WAS ORIGINALLY WRITTEN IN BASIC IN THE
; PEOPLE'S COMPUTER COMPANY NEWSPAPER VOL. 1, NO. 1
; (OCTOBER 1972)
;
;
0000 CK EQU 0DH ;CARRIAGE RETURN
000A LF EQU 0AH ;LINE FEED
0007 BEL EQU 07H ;BELL
;
; INITIALIZATION
;
0000 ORG 10H ;0-FH IS USED BY THE INTELEC MONITOR
0010 310004 INIT: LXI SP,0400H ;INITIALIZE STACK
;
; CALCULATE A TWO DIGIT RANDOM NUMBER
;
0013 C06D01 MNUM: CALL RND ;GET DECIMAL 0-9 IN (A)
0016 C02A01 CALL MPY10 ;(D) = 10 * (A)
0019 C06D01 CALL RND ;GET DECIMAL 0-9 IN (A)
001C 82 ADD D ;MAKE (A) DECIMAL 1-100
001D 325100 STA CNUM ;SAVE COMPUTER'S RANDOM NUMBER
0020 215200 LXI H,#10H ;COMPUTER HAS A NUMBER
0023 C06301 CALL MSG ;TELL THE USER
;
; GET TWO ASCII DECIMAL DIGITS FROM THE TELETYPE
;
0026 219600 GNUM: LXI H,#WYG ;ASK FOR NUMBERS
0029 C06301 CALL MSG
002C C03101 CALL DNUM ;(A) HAS A BINARY NUMBER FROM THE TTY
;
; DO THE RANDOM DIGITS AND THE GUESSES AGREE?
;
002F 215100 NCHK: LXI H,CNUM ;(H,L) POINTS TO THE COMPUTERS NUMBER
0032 8E CMP M ;COMPARE (A) AND ((H,L))
0033 CA4500 JZ WON ;EQUAL, A WINNER
0036 FA3F00 JM NCHK3 ;GUESS WAS TOO SMALL
0039 210D00 NCHK1: LXI H,#TS ;POINTER TO 'TOO BIG' MESSAGE
003C C34200 JMP NCHK4 ;GO OUTPUT 'TOO BIG' TO THE TTY
003F 210A00 NCHK3: LXI H,#TS ;POINTER TO 'TOO SMALL' MESSAGE
0042 C06301 NCHK4: CALL MSG ;OUT PUT MESSAGE TO TELETYPE
0045 C32600 JMP GNUM ;GO GET MORE GUESSES
;
; A WINNING NUMBER, CONGRATULATE THE USER
;
0048 21FF00 WON: LXI H,#CONG
004B C06301 CALL MSG ;CONGRATULATE THE USER
004E C31300 JMP MNUM ;GO COMPUTE ANOTHER NUMBER
;
; VARIABLES
;
0051 00 CNUM: DB 00H ;THE COMPUTERS RANDOM NUMBER
;
; MESSAGES
;
0052 0D0A0A #IHN: DB CR,LF,LF
0055 4920414D DB 'I AM THINKING OF A WHOLE NUMBER FROM 1 TO 100.'
0059 20544849
005D 4E4B494E
0061 47204F46
0065 20412057
0069 484F4C45
006D 204E554D
0071 42455220
0075 46524F4D
0079 20312054
007D 4F203130
0081 302E
0083 0D0A DB CR,LF
0085 47554553 DB 'GUESS MY NUMBER.'
0089 53204D59
008D 204E554D
0091 4245522E
0095 0D0A00 DB CR,LF,#0H
;
0098 0D0A0A #WYG: DB CR,LF,LF
009B 594F5552 DB 'YOUR GUESS? ',#0H
009F 20475545
00A3 53533F20
00A7 #0
;
00A8 2020204E #NEK: DB 'NUMBERS ONLY!',BEL,#0H
00AC 554D4245
00B0 5253204F
00B4 4E4C5921
00B8 0700
;
00BA 0D0A544F #TS: DB CR,LF,'TOO SMALL. TRY A BIGGER NUMBER.',#0H
00BE 4F20534D
00C2 414C4C2E
00C6 20205452
00CA 59204120
00CE 42494747
00D2 4552204E
00D6 554D4245
00DA 522E00
;
00DD 0D0A544F #TB: DB CR,LF,'TOO BIG. TRY A SMALLER NUMBER.',#0H
00E1 4F204249
00E5 472E2020
00E9 54525920
00ED 4120534D
00F1 414C4C45
00F5 52204E55
00F9 4D424552
00FD 2E00
;
00FF 07070700 #CONG: DB BEL,BEL,BEL,CR,LF
0103 0A DB 'YOU GUESSED IT!!! LETS PLAY AGAIN.'
0104 594F5552
0108 47554553
010C 53454420
0110 49542121
0114 21204C45
0118 5453205D
011C 4C415920
0120 41474147
0124 4E2E
0126 07070700 DB BEL,BEL,BEL,#0H

```

The 8080 assembly language program does the same thing that the Altair BASIC program elsewhere in this issue does. They do not look the same though. This is because the BASIC interpreter does things for you. It computes random numbers, converts digits input from the teletype into values in the BASIC variables, prints out messages, and conditionally branches depending on the result of a comparison. Each of these functions has to be done by the assembly language program.

Assembly language instructions for the 8080 system can perform the same functions that BASIC statements can. Quantities may be added, subtracted, or compared. It is necessary to have one quantity in the A register, and the other in another register or a memory location. The quantities are restricted to 8-bit integers. More complex routines may be written to handle 16-bit integers and even floating point quantities.

## Subroutines

If a certain process is to be done in several places, the code for it may be written once as a subroutine. This subroutine is then called from the places where you wish the process to occur. (Just like GOSUB in BASIC.) Subroutines have several advantages. Since the process is coded only once, space is saved. If you make any mistakes they only have to be corrected in one place. If you move the program to another computer with a different teletype controller, the teletype code only has to be changed in one place. Subroutines can separate processes into more understandable chunks. The main process in this program is comparing two numbers. By using subroutines, the conditional branches for teletype status, decimal input and actually comparing the numbers are separated from each other. Subroutines are self-contained, portable, already debugged packages. They can be put in any future programs. There is no reason to re-invent the wheel each time you write a program.

As an example of this, the subroutines that do teletype I/O on the Intelec 8 invert the data. To have this program work on an Altair or Processor Technology teletype controller, you have to change the CMA's into NOP's, and give the device addresses for the status and control instructions appropriate values for your system.

When the assembly language program compares each guess with the computer's random number, it sets flags depending on the result of the comparison. These flags are tested when the conditional jump, call, and return instructions are executed. If the zero flag is on (the quantities were equal, i.e. their difference was zero) the JZ instruction will be executed. The computer will jump to the WON statement. This is like an IF....THEN statement in BASIC. If the zero flag is not set, the next sequential instruction will be executed. If the sign bit is on (indicating that the quantity in the A register was less than that in the memory location), the JM instruction is executed. Otherwise the computer falls through to the next instruction.



SUBROUTINES

MULTIPLY THE CONTENTS OF (A) BY 10 AND PUT IT IN (D)

```

012A 07 MPY10: RLC          I(A) = (A) * 2
012B 57      MOV      D,A      I(D) = (A) * 2
012C 07      RLC          I(A) = (A) * 4
012D 07      RLC          I(A) = (A) * 8
012E 02      ADD      D      I(A) = (A) * 16
012F 37      MOV      D,A      I(D) = (A) * 16
0130 C9      RET          IRETURN
    
```

GET DIGITS FROM THE TELETYPE AND MAKE A BINARY NUMBER.  
 A CARRIAGE RETURN IS A VALID TERMINATOR, ALL OTHER NON-NUMERIC  
 CHARACTERS ARE ERRORS.

```

0131 0600 DNUM: MVI      B,0      IINITIALIZED TO ZERO
0133 CD9B01 DNI:  CALL    TTY1      IGET A CHARACTER FROM THE TELETYPE
0136 E67F      ANI      7FH      IGET RID OF THE PARITY BIT
0138 FE00      CPI      CR        IIS IT A CARRIAGE-RETURN
013A CA5701      JZ       DN2        IBRANCH IF SO
013D CDB101      CALL    TTY0        I Echo THE NUMBER
0140 DE30      SBI      30H      I CONVERT TO BINARY
0142 FA5901      JM      DN3        I TOO SMALL FOR A NUMBER
0145 FE0A      CPI      0AH      I
0147 F25901      JP      DN3        I TOO BIG FOR A NUMBER
014A 4F      MOV      C,A      I SAVE THIS DIGIT
014B 78      MOV      A,B      I GET THE PREVIOUS DIGITS
014C CD2A01      CALL    MPY10       IMULTIPLY BY 10
014F 81      ADD      C      I ADD IN THIS DIGIT
0150 47      MOV      B,A      I IT BECOMES THE OLD DIGIT
0151 DA3900      JC      NCHK1      I DO THIS IF IT OVER FLOWS
0154 C33301      JMP     DN1
    
```

MOVE THE ACCUMULATED DIGIT TO (A) AND RETURN

```

0157 76 DN2:  MOV      A,B      I(A) RETURNS WITH THE NUMBER
0158 C9      RET
    
```

ONLY NUMBERS ARE VALID INPUT

```

0159 21A000 DN3:  LXI      H,0NEH      I POINTER TO ERROR MESSAGE
015C CD6301      CALL    MSG          I OUTPUT TO TELETYPE
015F E1      POP      H          I RESTORE STACK POINTER
0160 C32600      JMP     GNUM        I GO ASK FOR NUMBERS AGAIN
    
```

MSG OUTPUTS AN ASCII STRING TO THE CONSOLE.  
 (HL) POINTS TO THE FIRST CHARACTER OF THE MESSAGE  
 CHARACTERS ARE OUTPUT UNTIL A BINARY ZERO IS ENCOUNTERED,  
 WHICH TERMINATES THE MESSAGE.

```

0163 7E MSG:  MOV      A,M      I LOAD A CHARACTER
0164 87      ORA      A          I IS IT A BINARY ZERO?
0165 C8      RZ          I IF SO, RETURN
0166 CDB101      CALL    TTY0        I OUTPUT (A) TO TELETYPE
0169 23      INX      H          I (HL) POINTS TO THE NEXT CHARACTER
016A C36301      JMP     MSG        I TRY THE NEXT CHARACTER
    
```

RND RANDOM NUMBER GENERATOR  
 THIS IS TAKEN FROM THE COMPUTER HOBBYIST VOL 1, NUMBER 5  
 JIM PANKER WROTE IT.

```

016D 219A01 RND:  LXI      H,SHIFT+3
0170 0600      MVI      B,8D
0172 7E      MOV      A,M
0173 07 RND1:  RLC
0174 07      RLC
0175 07      RLC
0176 AE      XRA      M
0177 17      RAL
0178 17      RAL
0179 219701      LXI      H,SHIFT
017C 7E      MOV      A,M
017D 17      RAL
017E 77      MOV      M,A
017F 23      INX      H
0180 7E      MOV      A,M
0181 17      RAL
0182 77      MOV      M,A
0183 23      INX      H
0184 7E      MOV      A,M
0185 17      RAL
0186 77      MOV      M,A
0187 23      INX      H
0188 7E      MOV      A,M
0189 17      RAL
018A 77      MOV      M,A
018B 05      DCR      B
018C C27301      JNZ     RND1
018F E60F      ANI      0FH      I ISOLATE BOTTOM 4 BITS
0191 FE0A      CPI      10D      I COMPARE WITH 10D
0193 F26D01      JP      RND        I TOO BIG, GET ANOTHER RANDOM NUMBER
0196 C9      RET
0197 A1B2C304 SHIFT: DB 0A1H,0B2H,0C3H,0D4H
    
```

TTY1 TELETYPE INPUT ROUTINE  
 THIS ROUTINE ALSO INCREMENTS ONE OF THE VARIABLES IN THE  
 RANDOM NUMBER GENERATOR

```

019B E5 TTY1:  PUSH    H          I SAVE (HL)
019C 219A01      LXI      H,SHIFT+3      I (HL) POINTS TO A RANDOM NUMBER VARIABLE
019F 3E03      MVI      A,3H
01A1 DB01      IN       TTYSP          I TAKE TELETYPE STATUS
01A3 E601      ANI      TTYDA          I CHECK IF DATA IS AVAILABLE
01A5 CAAC01      JZ       TTY12         I YES, GO GET IT
01A8 34      INX      M          I INCREMENT THE RANDOM NUMBER VARIABLE
01A9 C39F01      JMP     TTY11         I LOOP UNTIL YOU GET DATA AVAILABLE
01AC DB00      TTY12:  IN       TTYIP          I INPUT THE CHARACTER
01AE 2F      CMA          I INVERT IT FOR INTEL
01AF E1      POP     H          I RESTORE THE STACK POINTER
01B0 C9      RET          I RETURN
0001      TTYDA  EQU    1          I TELETYPE DATA AVAILABLE
0000      TTYIP  EQU    0          I TELETYPE INPUT PORT ADDRESS
    
```

TTY0 TELETYPE OUTPUT ROUTINE

```

01B1 4F TTY0:  MOV      C,A      I SAVE THE CHARACTER
01B2 3E03      MVI      A,3H
01B4 DB01      IN       TTYSP          I TAKE TELETYPE STATUS
01B6 E604      ANI      TTYBE          I
01B8 C2B201      JNZ     TTYO1         I LOOP UNTIL IT IS READY
01BB 79      MOV      A,C          I PUT THE CHARACTER IN (A)
01BC 2F      CMA          I INVERT IT FOR INTEL
01BD D300      OUT     TTYOP          I OUTPUT THE CHARACTER
01BF 79      MOV      A,C          I GET THE CHARACTER IN (A)
01C0 C9      RET          I RETURN
0001      TTYSP  EQU    1          I TELETYPE STATUS
0004      TTYBE  EQU    4          I TELETYPE BUFFER EMPTY
0000      TTYOP  EQU    0          I TELETYPE OUTPUT ADDRESS
0000      END
    
```

# Stack

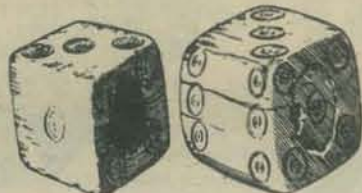
Several of the 8080 instructions manipulate a thing called the stack. The computer can write itself memos and put them in its stack. The memos are not written on pieces of paper, they are put into two consecutive words of its memory. The computer can recover the memos in the reverse order that they were put on the stack. This is like putting messages in a pile and being able to recover the top one and then the previous one etc.. This is called a LIFO or last-in-first-out stack. The 8080 has a sixteen bit register called a stack pointer, or SP for short. It contains an address of a memory location. When 16 bits of data is put on the stack, 8 bits of it is put in the memory location whose address is in the stack pointer. The second 8 bits is put in the next lower memory location. Two is subtracted from the address in the stack pointer. When 16 bits of data is taken off the stack, 8 bits comes from the word whose address is one greater than the address in the stack pointer. 8 bits come from the word whose address is two greater than the address in the stack pointer. The address in the stack pointer is then increased by two.

# Where to get it

Several (as far as we know) text-editor assembler systems are available to the hobbyist. MITS has the "Altair Package One". Using this system, you can create a source program and assemble it. This costs \$30.00 for Altair owners and \$175.00 for others. When ordering, specify paper-tape or cassette.

Processor Technology sells documentation for its text-editor assembler system for \$3.00. They distribute paper-tapes of the object program through local computer clubs. Write to them if your club does not have a copy! This system creates a source program in memory, assembles it, and puts the object program in memory too. An expanded version of this system can keep files on external media. It can have up to six files in memory and assign names to them. It can assemble files from the external media or from memory. It can chain common symbols from one program to another. This system is on PROMS and costs \$250.00. They also have an 8080 simulator that can be added to the above system. It costs \$95.00.

A different approach to programming is a system distributed by Robert Swartz. This system lets the user enter a program by typing in the mnemonics. The user may start execution at an address he specifies. Breakpoints may be put in the program. A breakpoint will stop the machine and save the contents of the registers. By examining the contents of the registers, the user may tell why the program is not doing what he wanted it to do. If the program does not stop at a breakpoint, it tells the user something too. Perhaps some conditional branch was not set up correctly. Sections of code may be relocated. This adjusts the addresses in memory-reference instructions. Sections of code may be copied with no changes too. Documentation, instructions, and a source listing costs \$20.00. A paper-tape of the object program costs \$20.00. This guy delivers quick! This system is well worth the cost!



Processor Technology  
 2465 Fourth St.  
 Berkeley, CA 94710

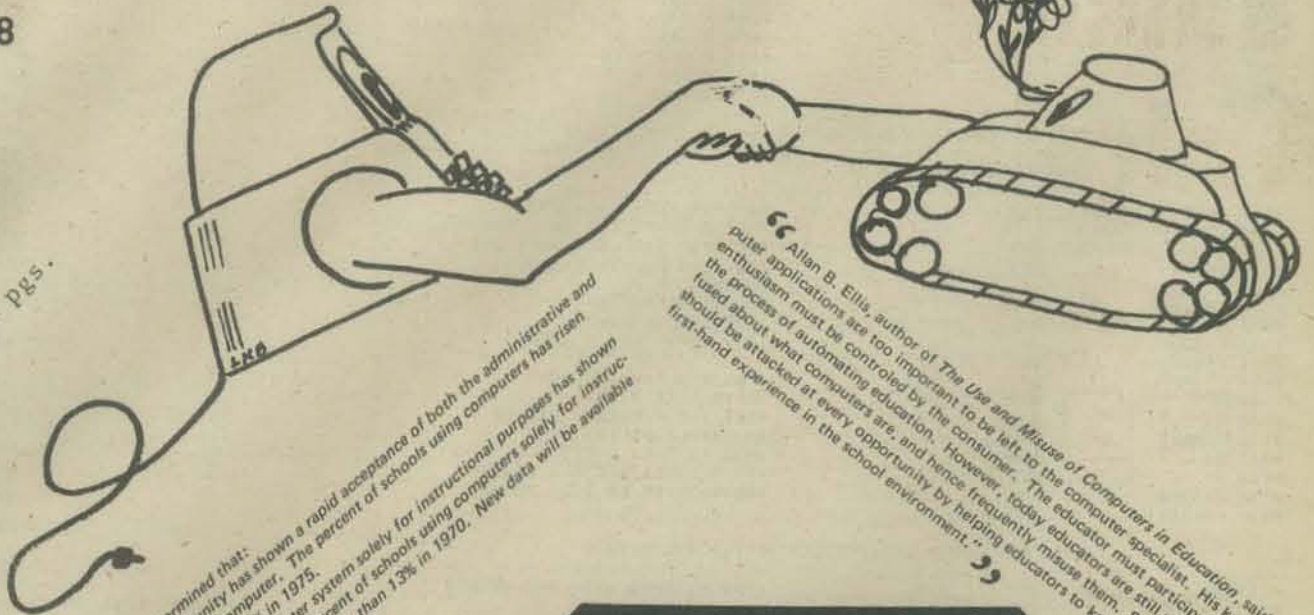
Robert Swartz  
 195 Ivy Lane  
 Highland Park, IL 60035

MITS  
 6328 Linn NE  
 Albuquerque, NM 87108

15 JUNE 1975

# THE SCHOOL COMPUTER

62 pgs.



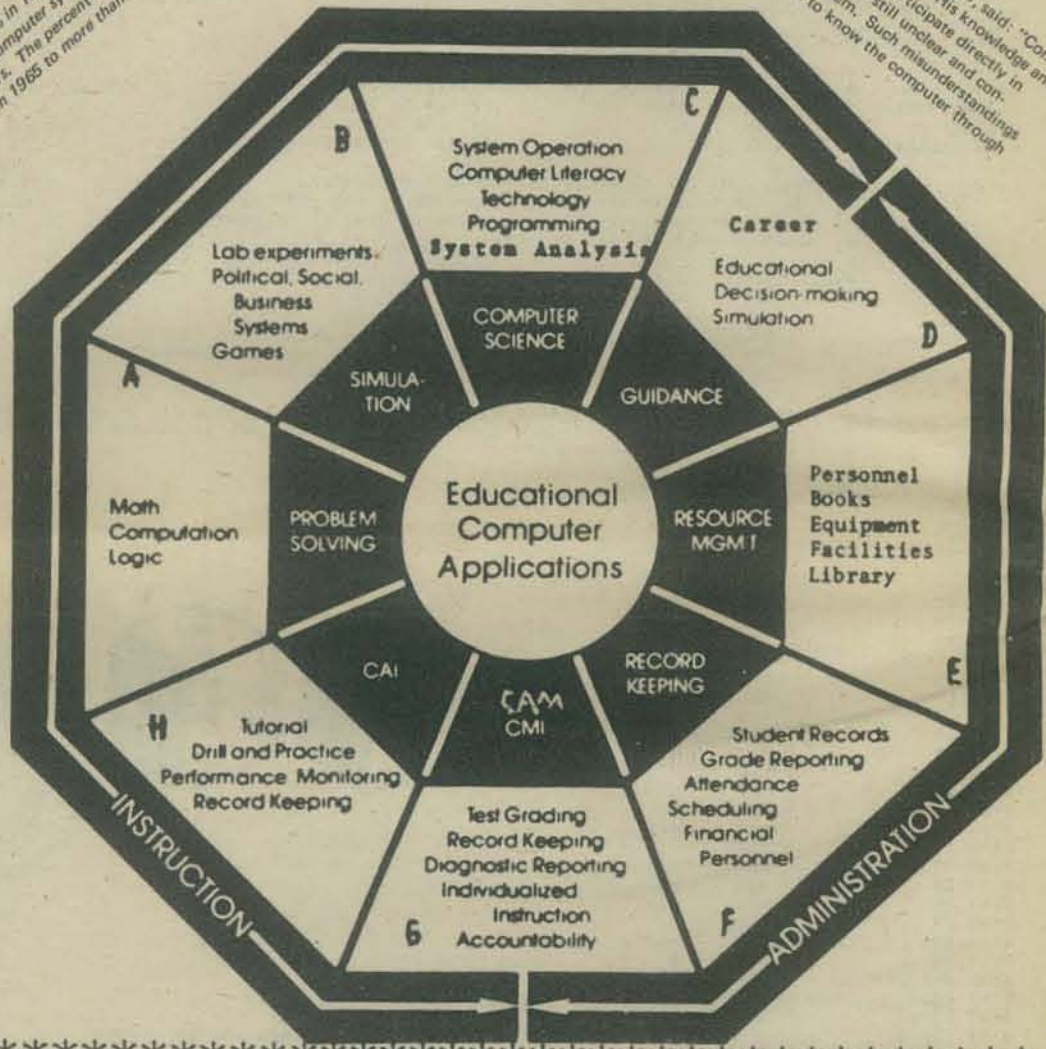
“ Allan B. Ellis, author of *The Use and Misuse of Computers in Education*, said: “Computer applications are too important to be left to the computer specialist. His knowledge and enthusiasm must be controlled by the consumer. However, today educators are still unclear and confused about what automating education is, and hence frequently misuse them. Such misunderstandings should be attacked at every opportunity by helping educators to know the computer through first-hand experience in the school environment.”

- “ A review of the anticipated tangible and intangible instructional benefits are:
- a. Substantially improve learning among both slow and fast learners.
  - b. Provide unprecedented motivation.
  - c. Provide experience with a vital, contemporary machine.
  - d. Enrich existing programs in the mathematics and sciences through the use of the school computing center.
  - e. Foster student appreciation of the computer's role in society.
  - f. Remove the mystery and bewilderment that exists in many of the students' minds about computers and automation.
  - g. Provide students the opportunity to work on creative and complex problems that would be impossible to solve by manual methods.
  - h. Foster discipline, clarity, and methodology in problem solving.
  - i. Teach algorithmic thinking and explore rigorous thought processes.
  - j. Provide motivation and more individual, creatively challenging instruction in a variety of curriculum areas.
  - k. Provide teachers with a versatile aid to regular class instruction.
  - l. Prepare college bound students with an understanding of the computer and how it can and cannot be used to solve problems.
  - m. Provide experience with career preparation in a field which offers challenge and high compensation.
  - n. Develop an early recognition of a deeper insight into the concepts of mathematics, logic, and science.”

“ The AIR Survey determined that:

- a. The educational community has shown a rapid acceptance of both the administrative and instructional capabilities of the computer. The percent of schools using computers has risen from 35% in 1970 to more than 45% in 1975.
- b. The use of an educational computer system solely for instructional purposes has shown a growth rate in the last few years. The percent of schools using computers solely for instructional purposes rose from 2% in 1965 to more than 13% in 1970. New data will be available in late 1975.”

AIR = American Institutes for Research (While we were preparing this page we received COMPUTING ACTIVITIES IN SECONDARY EDUCATION (Final Report) from AIR, 165 pages of extensive, excellent research! AIR's at 3301 New Mexico Av., Wash., DC 20016 — neither this nor The School Computer is priced — write 'em for info)



## TABLE OF CONTENTS

- 1 USDESEA And The School Computer  
*The USDESEA experience in automatic data processing* ||| *USDESEA secondary schools utilizing data processing* ||| *The USDESEA objectives in automatic data processing*
- 2 Why Computers in the Classroom?  
*Introductory questions* ||| *Preliminary Answers* ||| *Growth Rate of computers in schools* ||| *The Computer as pupil* ||| *Evaluation of the Dartmouth Project* ||| *Answers to questions* ||| *Conclusions*
- 3 Preparing USDESEA teachers and administrators to use the School Computer  
*Introduction* ||| *Overview* ||| *Course I: General introduction for administrators and teachers* ||| *Course II: for administrators* ||| *Course II: for teachers -- mathematics applications* | *Science applications* | *Business Education applications* | *English applications* | *Social Science applications* | *Computer Technology* ||| *Course III: for administrators* ||| *Course III: for teachers*
- 4 Installation and Operation Guidelines  
*Computer site facility considerations* ||| *USDESEA computer coordinators and Darmstadt Career Center support functions* ||| *Local school coordinators and their functions* ||| *Other educational agency use* ||| *Computer supplies and curriculum materials* ||| *The INTERDATA 7/16 System and its operational procedures* ||| *Evaluation of the USDESEA school computer project*

## APPENDICES

- SCHOOL COMPUTER COORDINATORS ||| THE SCHOOL COMPUTER LOG AND MAINTENANCE ||| SUPERVISORY SERVICES AGREEMENT ||| USAGE PROFILE ||| SCHOOL COMPUTER LIBRARY PROGRAM DESCRIPTION ||| COMPUTERS AND SOCIETY



DIRECTORATE  
DEPARTMENT OF THE ARMY  
UNITED STATES DEPENDENT SCHOOLS, EUROPEAN AREA  
APO 09175

\*\*\*\*\*  
SUBJECT: Information Concerning DoD Schools Computer Project

TO: Editor Peoples Computer Company

- 1) Eighteen months ago the United States Army helped the United States Dependent Schools, European Area (USDESEA) to launch a broad and comprehensive program to involve the School Computer in the instructional process. The U.S. Army selected the computer model, Interdata 7/16, a very powerful unit capable of performing services far in excess of applications in the instructional program. Computers were first installed in 13 high schools, currently in 23 high schools, and soon in 35 high schools. Current application areas include Problem Solving, Simulations, and Computer Technology.
- 2) Current negotiations are for the provision of support for student guidance services; student scheduling and record-keeping systems, material, supply, and personal accounting programs; and assessment monitors of progress towards educational goals. A pilot system consisting of card reader, CRT, line printer, 10 megabyte disk, and high speed paper tape reader/punch is now at Darmstadt Career Center in Germany.
- 3) USDESEA now has almost ten years of in-depth computer experience not only on many large military computers, but especially with the Interdata 7/16. The benefits to the system are fourfold:
  - a. From the socio-economic viewpoint, the School Computer Project provides the participants with comprehensive knowledge of electronic data processing which facilitates employment opportunities for graduates of USDESEA.
  - b. From the academic aspect, the program provides participants with the workable ADP knowledge required by almost all higher educational endeavor.
  - c. From the instructional side, the School Computer Project fosters discipline and clarity in problem solving, provides teachers with a versatile aid to regular classroom instruction and provide experience with dynamic contemporary equipment.
  - d. For school management, the School Computer Project in its augmented form will provide timely information from which decisions can be derived to provide optimum resource utilization.
  - e. For information concerning employment with USDESEA, write:

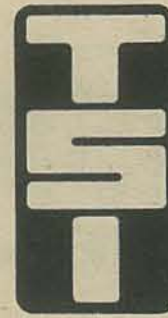
Deputy Chief of Staff for Personnel  
Dependents Schools Teacher Recruitment Branch  
Department of the Army  
The Pentagon Room 1-A-658  
Washington, D.C. 20310



# Talking Calculator

The TALKING CALCULATOR is here. It is called SPEECH PLUS\* and costs \$395.

It is designed for use by the blind — but, who knows how it may be used? Perhaps to teach arithmetic to young children.



**TELESENSORY  
SYSTEMS, INC.**


1889 Page Mill Road  
Palo Alto, California  
94304

415 - 493 - 2626

Press these keys



and SPEECH PLUS\* will say "CLEAR THREE PLUS FOUR EQUALS"

Then press  and you will hear . . .

"SEVEN POINT OH OH"



SPEECH PLUS\* can pronounce every one of the 24 keys on the keyboard. Use a tape recorder to make a complete record of your "conversation." Turn the volume up or down to suit yourself. For completely private listening, an earphone is included. For info write, 1889 Page Mill Rd, Palo Alto, Ca. 94304

Let your fingers  
do the talking —

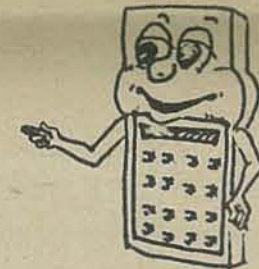
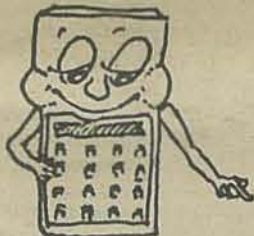


\*SPEECH PLUS is a trademark of Telesensory Systems, Inc.

**AND NOW...THE FIRST TALKING CALCULATOR GAME!**

## Say Something Special

by Sivasailam Thiagarajan and Harold D. Stolovitch



Can you make the calculator say the special number when you press the "speak" key? We hope so, because your score depends on the number of times you can make the calculator repeat this special number.

**NUMBER OF PLAYERS:** Two to five.

**APPROXIMATE TIME REQUIREMENT:** Three to five minutes.

**SKILLS INVOLVED:** The four basic operations.

**CHANCE FACTOR:** Some. However, in a series of games, things balance themselves.

**PLAY OF THE GAME:**

(1) One player punches in a four-digit number less than 5000. The next player adds another four-digit number under 5000 to it. He then presses the equals and "speak" keys to hear the sum.

*Valerie punches in 1821 and Paula adds 2563. When the "speak" key is pressed, the calculator reports, "Four . . . three . . . eight . . . four."*

(2) The first digit of the sum is the special number. The second digit is the working number. The last two digits make up the base number.

*Since the sum was 4384, 4 is the special number; three is the working number; and 84 is the base number.*

(3) The object of the game is to add, subtract, multiply or divide the working number with the base number so that the special number occurs in the result. If this is not possible, the player performs any one of the four operations to create a new base number and passes the calculator to the next player.

*Jeff, the third player, tries all possible operations with the 3 and the 84 in his head but can't figure out a way to get the special number 4 in the answer. In frustration, he settles on 84 divided by 3. The calculator says, "Two . . . eight."*

(4) Game continues in the same way until one player makes the calculator say "something special" in the result. The player's score then, is the number of times the calculator repeats the special number.

*Valerie thinks about the 28 which Jeff gave her. This is her base number. She too can't do anything with it and the working number 3. In disgust she adds 3 and the calculator intones the new base number "Three . . . one."*

*Paula is delighted. She snatches the calculator and quickly adds 3 to 31. The calculator calls out, "Three . . . four." Paula gets one point for making the calculator say the special number once. She wins the round.*

(5) Game is repeated until one player accumulates a score of 10 points.

*During the next game, the players begin with 3325 as the four-digit sum. This makes the 3 both the special number and the working number. Jeff takes the base number of 25 and divides it by the 3. He is rewarded by the calculator declaring, "Eight point three, three, three, three, three, three." With his brilliant move Jeff jumps to the lead with a score of seven points because the calculator repeated the special number seven times.*

*In later games, however, he is not so lucky. Valerie comes from behind and gets to the winning total of 10 points.*

# TINY BASIC

by DENNIS ALLISON, Bernard Greening,  & lots of FRIENDS!!



Dear People,

After a quick pique at TINY BASIC I have the following (possibly ill-considered) comments:

1. It looks useful for tiny computers, which is as intended.
2. Those accustomed to extended BASIC, or even the original Dartmouth BASIC, will be irked by its limitations. But then, that's how the bits byte!
3. How does the interpreter scan the word THEN in an IF statement?
4. Some of the comments for EXPR seem to be on the wrong line, or my reading is more biased than usual.
5. Users should note that arithmetic expressions are evaluated left-to-right unless subexpressions are parenthesized (i.e., there is no implicit operator precedence).
6. Real numbers would be nice, but would take up a lot more space. Probably too much. Ditto for arrays and string variables.
7. Please consider adding semicolon (i.e., unzoned) PRINT format with a trailing semicolon inhibiting the CRLF. This would be very useful and would be easy to add.
8. If INPUT prompts with a question mark, please print a blank character after the question mark (for readability).
9. I suggest allowing THEN as a separator in any multi-statement line, not just in IF statements. Since lines like  

```
IF 5<X THEN IF X<10 THEN GOSUB 100
```

are already legal, why not allow lines like  

```
LET A=B THEN PRINT A
```

or any other combination, including silly ones like  

```
GOTO 200 THEN INPUT Z
```

the second statement of which would never be executed. If THEN works for IF, it should be possible to make it work for anything.
10. I also suggest allowing comments somehow. At present, comments must be held to a minimum  

```
IF X<>X THEN PRINT "THIS IS A COMMENT"
```

are possible via subterfuges such as  

```
IF X<>X THEN PRINT "THIS IS A COMMENT"
```

but that seems kind of gauche. Naturally, comments must be held to a minimum in TINY BASIC, but sometimes they may be vital.
11. Doing a  

```
PRINT " "
```

seems to be the only way to print a blank line. Well, all right.
12. Exponentiation via \*\* would seem fairly easy to add, and might be worthwhile.
13. By the way, all of this will execute in 1K, won't it?

Jim Day  
17042 Gunther St.  
Granada Hills, CA 91344

Answering your Questions by number where appropriate:

- 3&4. Woops! There should be a TST instruction to scan the THEN. The comments are displaced a line. See the corrected IL listing in this issue.
5. Expressions are evaluated left-to-right with operator precedence. That is,  $3+2*5$  gives 13 and not 25. To see this, note that the routine EXPR which handles addition gets the operands onto the stack before returning.
7. Agreed, but this is intended as a minimal system.
9. One man's syntactic sugar is another's poison. I don't like the idea. Incidentally, how would you interpret  

```
LET A=B THEN GOSUB 200 THEN PRINT 'A'
```

The GOSUB then has to store a program address which botches up the line entry routine or one has to zap the GOSUB stack when an error is found. Both are solved only by kludges.
- 10-12. See 7.
13. Maybe. But 2K certainly. See below.



Dear PCC,

I am thrilled with your idea of an IL but I think that if you intend only to write a BASIC interpreter that a good symbolic assembler would be appropriate. With an assembler similar to DEC's PAL 3 or PAL 8 the necessary routines could be written and used in nearly the same way without having to write the associated run time material that would be necessary for its use as an interpreter. A command decoder, a text buffer, and a line editor would be necessary and all of this uses up a good amount of space in memory.

If you are aware of all these things and still plan to develop an IL interpreter, then I suggest you start as DEC did with a simple symbolic editor as the backbone of the interpreter. In

this way you allow a 2800% increase in development and debugging speed (according to Datamation's comparison of interpreters and compilers whose fundamental difference is the on line editing capability). Once this has been implemented and IL is running on a particular system then the development of interpreters of all types is greatly simplified. By suggesting IL you have stumbled onto the most logical and easiest way to develop a complete library of interpreters. In addition to BASIC, it is very easy to write interpreters for: FOCAL, ALGOL, FORTRAN, PL 1, LISP, COBOL, SNOWBAL, PL/m, APL, and develop custom interpreters with the ease with which one would write a long BASIC program!

As I pointed out earlier, all these features take up memory space and, as you have pointed out, run time is much slower. The way around this is to define the IL commands in assembly language subroutines then assemble the completed interpreter as calls to these subroutines. Thus the need for the IL interpreter as a run time space and time consumer is no longer necessary! (OK symbolic assembler haters, let's see you do this in machine language in less than ten man-years!)

In places where time and space are not so much of a problem, I suggest the addition of an interrupt handler and priority scheduler to allow IL to be used as a simplified and painless TIMESHARED system enabling many users to run in an interpreter and use more than one interpreter at once. Multi-lingual timeshare systems previously being available to those who have a highspeed swapping disk, drum, or virtual memory, are now available to the user who has about 16K of memory and a method of equitably bringing interpreters in to main memory from the outside world (a paper tape reader or cassette system is the easiest to come by).

In short, IL as I suggested, in its minor stages would be a powerful software development aid; and in its final, most complex stages would provide a runtime system of unheard of inexpensive.

I have heard from unofficial sources that ordinarily an interpreter or compiler requires ten man-years to write and debug to the point of use (if one man works the job would require 10 years, if 10 men work it would take one year). Since this is to be expected as the initial development of IL and since I have a general idea of the circulation of PCC, we should have IL up and running by the next issue of PCC!

At this time I would like to request a few reprints of the article dealing with IL because I want to get some help from others in my school in getting a timeshared version working on our 16K PDP 8/m with DECTAPE. I seem to have lent my copy of that issue to one of the people I had been trying to get on this project and he has not returned it to me. Meanwhile, I need the article to begin initial work on the interpreter to insure compatibility with the version coming across through PCC. I will keep you posted as with regards to the development.

William Cattey  
39 Pequet Road  
Wallingford, Ct. 06492

The IL approach to implementation is quite standard and dates back to Schorre's META II, Gleenie's Syntax Machine, and numerous early compilers. It was widely used in the Digitek FORTRAN systems. We did not "stumble" on to the technique, we chose it with some deliberation.

You are right that a symbolic assembler can be used either to assemble the pseudocode into an appropriate form or to expand the pseudocode into actual machine instructions with the attendant cost in space (and decrease in execution time). Our goal is a small, easily transportable system. The interpretive approach seems consistent with this primary goal. We are using the Intel 8080 assembler's macro facility to assemble our pseudocode.

I certainly agree that it is relatively easy (but not simple!) to implement other languages using the IL approach. From the users standpoint, provided he is not compute bound, there is little difference. Interpreters are often a bit more forgiving of errors and can give better diagnostics.

In my experience, your figure of 10 man-years is high for some languages and low for others. A figure of two to four man-years is probably more accurate, and that includes documentation at both the implementation and user level. Good luck on your implementation.

...I have found in my adaptation of it (TINY BASIC IL) for full use that certain commands need strengthening, while some might be dropped. I will hopefully be coming out with these possible modifications. Concerning my ideas on space trade-offs; I think an assembled version would take less space, since each command

is treated as a subroutine call in a program made up of routines, while the interpreter needs a run time system in the background which, since it is interpretive in itself, takes up space.

P.S. You missed my allusion to assembler over strictly octal or hexadecimal op codes (my meaning was twofold). In DEC's PAL8 assembler the following syntax is needed to make the most efficient use of routine calling:

```
TSTN=JMSI (jump to subroutine indirectly via this location)
100 XTSTN
```

The assembler shows the binary as if TSTN were like a JMSI 100/JAP to subroutine indirectly via 100 (requiring very very little extra space per routine—one word, to be exact).

I would be happy to resolve any questions regarding compilers vs. interpreters. (Datamation did an article on the writing of a standard program in several languages then documented development and run time.)

William Cattey

There are several different varieties of interpreters. One is simply a sequence of subroutine calls. Another is, as you suggest, a list of indirect references to subroutine calls. We are considering a different organization where the call address and some additional information is packed into a single byte. This is a good strategy vis a vis memory conservation only if the size of the code memory to decode the packed instruction plus the size of the encoded instructions is smaller than the size of a more straightforward encoding. This remains to be seen.

I guess I did miss your point on assemblers. However, let me assure you that I would never advocate making software by programming directly in hex or binary. Even an assembler seems cumbersome and difficult to me; I prefer a good systems language like PL/M!



Dear Dennis and other PCCers,

In my last crazy jumbled letter I made some comments about TINY BASIC. Here is the result of 2-3 days work and thinking about it. Instead of having an interpretive IL, I chose to set it up as detailed as possible, then have people with different machines code up subroutines to perform each IL instruction. I'm not convinced that this way would take more space, and I'm sure it would be faster.

There are a couple of changes in the syntax from your published version: separate commands from statements, add terminal comma to PRINT, and restrict IF-THEN to a line number (implied GOTO).

The semantics are separated out from the syntax in IL as much as possible. This should make it easier to be clear about what the results of any given syntactic structure. This is most apparent in the TST instructions, and the elimination of the NXT instruction. That one in particular was a confusion.

Please let me know how this fits with what you're doing. I don't have a micro yet—time, not money prevents it.

I was very pleased with MITS seminar in Boston. I went on Oct. 27. Only hassle was that they "forgot" to include a current catalog and price list in the materials. There were about 60 people there, with more expected the next night. I met some wonderful people, found out about a micro club in Reading (just north of Boston), and that a MITS store will open soon in the area.

Have been looking at all kinds of micro stuff. Found another magazine with good articles called EDN (formerly Electronic Design News). They have a free distribution set up, but its very restricted, so the \$20/year isn't bad—for 24 issues. The subscription address is:

EDN  
270 St. Paul St.  
Denver, Col. 80206.

John Rible  
51 Davenport St.  
Cambridge, MA 02140

Because of space limitations, we have not been able to publish all of John Rible's version (dialect) of TINY BASIC. We'll probably include it in the first issue of the TINY BASIC NEWSLETTER.

By separating the syntax from the semantics he has produced a larger and possibly simpler to understand IL. There are more IL instructions so, I believe, the resultant system will be larger; further, the speed of execution is roughly proportional to the number of IL instructions (decoding IL is costly), it will be slower.

# EXTENDABLE TINY BASIC

JOHN RIBLE

## INTERMEDIATE LANGUAGE PHILOSOPHY

Instead of IL being interpreted, my goal has been to describe IL well enough that almost anyone will be able to code the instructions as either single machine language instructions or small subroutines. Besides speeding up TINY BASIC, this should decrease its size. Most of the instructions are similar to those of Dennis' (PCC V4 no. 2), but the syntactical has been separated from the active routines. This would be useful if you want the syntax errors to be printed while inputting the line, rather than when RUNNING the program.

Most subroutines (STMT, EXPR, etc.) are recursively called, so in addition to the return address being stacked, all the related data must be stacked. This can use up space quickly.



## SYNTAX for John Rible's version of TINY BASIC

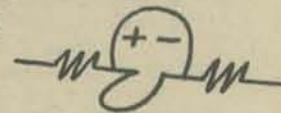
```

<PROGRAM> ::= <PLINE>*1
<PLINE> ::= <NUMBER> <STATEMENT>
<ILINE> ::= <COMMAND> | <STATEMENT>
<COMMAND> ::= CLEAR | LIST | RUN
<STATEMENT> ::=
    @ |
    LET <VAR> = <EXPR> @ |
    GOTO <EXPR> @ |
    GOSUB <EXPR> @ |
    PRINT <EXPR-LIST> (, | ε) @ |
    IF <EXPR> <RELOP> <EXPR>
        THEN <STATEMENT> @ |
    INPUT <VAR-LIST> @ |
    RETURN @ |
    END @
<EXPR-LIST> ::= (<STRING> | <EXPR>) (, | <STRING> | <EXPR>)*2
<STRING> ::= " <ANY CHAR> *2"
<ANY-CHAR> ::= any character except " or @
<EXPR> ::= (+ | - | ε) <TERM> ( + | - ) <TERM>)*2
<TERM> ::= <FACTOR> ( ( * | / ) <FACTOR>)*2
<FACTOR> ::= <VAR> | <NUMBER> | <EXPR>
<VAR-LIST> ::= <VAR> (, | <VAR>)*2
<VAR> ::= A | B | ... | Y | Z
<NUMBER> ::= <DIGIT> <DIGIT>*3
<DIGIT> ::= 0 | 1 | ... | 8 | 9
<RELOP> ::= < ( = | > | ε) | < ( = | < | ε) =
  
```

notes: ε is null character  
actual characters are in bold face  
\*1 repeat limited by size of program memory space  
\*2 repeat limited by length of line  
\*3 repeated 0 to 4 times



Yes, BASIC programs for BAUDOT TTY do look a little different from BASIC programs written for ASCII TTY. But you can figure out the differences!



Dear Mr. Allison,

I was very interested in your Tiny BASIC article in PCC. Your ideas seem quite good. I have a few suggestions regarding your IL system. I hope I am not being presumptuous or premature with this. Unless I misunderstood you, your IL encoding scheme seems inadequate. For instance, IL Jumps must be capable of going up and down from the current PC. This means allotting one of the 6 remaining bits of the IL byte as a sign bit resulting in a maximum PC change of +31 which is not adequate in some cases, ie. the JMP from just above S17 back to START. May I suggest the following scheme which is based on 2 bytes per IL instruction:

IL		ML	
JMP	CALL	TST	CALL
OXX <sub>8</sub>	1XX <sub>8</sub>	2XX <sub>8</sub>	1XX <sub>8</sub> (1st byte)
YYY <sub>8</sub>	YYY <sub>8</sub>	YYY <sub>8</sub>	YYY <sub>8</sub> (2nd byte)

where XX= lower 6 bits of high part of address (assume upper 2 bits are 00)

YYY= all 8 bits of low part of address.

The complete address being OXXYYY<sub>8</sub>. These addresses represent the locations associated with the IL and ML instructions. Note that if X points to a table with a stored address, you have 3 bytes used—my scheme uses only 2 bytes with the same basic information.

I also wondered about the TST character string. In my implementation I am using the following technique: the string follows the TST byte pair immediately with a bit 7 set in the last character.

Example: 240 } TST fail address in 040006<sub>8</sub>  
006 }  
0 {L}  
0 {E}  
1 {T}

On the TSTL, TSTV, and TSTN IL's, it appears you need a ML address for the particular subroutine and 2 additional bytes for the fail address. At least this is how I am handling it.

I am looking forward to future articles in the series.

Thanks again—keep up the good work!

P.S. I am co-owner of an Altair. We are writing our Tiny BASIC in Baudot to feed our Model 19's.

Richard Whipple  
305-Clemson Dr.  
Tyler, Tx. 75701

We found the same problem with the published IL interpreter. We solved it by doing a bit of rearranging and introducing a new operations code which does jumps relative to the start of the program, but has the same basic encoding. Your mechanization will, of course, work, but requires one more byte per IL instruction, may be harder to implement on some machines, and takes more code.

We are using the same scheme of string termination (i.e., using the parity bit) as you are. It's simple, easy to test, and difficult to get into the assembler.

There are a few errors and oversights in the IL language and in the interpreter you didn't mention. See the new listing in this issue.

Good luck. Keep us informed of your progress.

Thanks for your reply. I suppose you have received an additional letter containing my program runs in TINY BASIC. Sorry I did not provide any explanation with the last letter but I was in a state of excitement and elation at the time and just shot that letter off—I am writing a letter to you at this time with further comments about the programs and particularly our use of BAUDOT. But I will get this reply off immediately so you will not be held up. By the way, I am co-owner of an Altair with John Arnold who is on your subscription list.



Hi computer buffs—TINY BASIC is up and running! Have a look. We're BAUDOT nuts here—that's why the symbols look a little strange. The TINY BASIC interpreter fits in about 2K. If we can assist you in some way, drop us a line. We remain happy Altair owners—

Dick Whipple  
John Arnold  
Rt. 4 Box 52A  
Tyler, TX 75701

BAUDOT is a code like ASC II, but it is five bit characters, not eight. To get a full character set, it has an escape character to cause an upshift and a downshift.

## :LIST

```

10 LET X:1
20 PRINT "N", "N SQUARED", "N CUBED"
30 PRINT X, X#X, X#X#X
40 LET X:X&1
50 IF X:1 END
60 GOTO 30
:RUN
  
```

N	N SQUARED	N CUBED
1	1	1
2	4	8
3	9	27
4	16	64
5	25	125
6	36	216
7	49	343
8	64	512
9	81	729
10	100	1000

## :LIST

```

10 PRINT "NUMBER TO BE FACTORED";!
20 INPUT N
30 PRINT ""
130 IF N:0 GO TO 999
140 IF N:2 GO TO 250
150 LET Z:0
170 LET F:2
180 GOSUB 310
185 IF N:1 GO TO 500
190 LET F:3
200 GOSUB 310
205 IF N:1 GO TO 500
206 LET F:F&2
207 IF F:(N GOTO 200
220 IF Z:0 GO TO 250
230 PRINT N;!
240 GO TO 500
250 PRINT N;" IS PRIME"
260 GO TO 500
310 IF N/F#F-N) ( 0 GOTO 400
330 IF Z)0 GO TO 370
340 PRINT N;" HAS THE FACTORS : "
350 PRINT " " ;!
360 LET Z:1
370 PRINT F;!
380 LET N:N/F
390 IF N) ( 1 GO TO 310
400 RETURN
500 PRINT""
510 PRINT""
530 PRINT""
540 GO TO 10
999 END
:RUN
  
```

NUMBER TO BE FACTORED ?24  
24 HAS THE FACTORS :  
2 2 2 3

NUMBER TO BE FACTORED ?37  
37 IS PRIME

NUMBER TO BE FACTORED ?1365  
1365 HAS THE FACTORS :  
3 5 7 13

NUMBER TO BE FACTORED ?256  
256 HAS THE FACTORS :  
2 2 2 2 2 2 2 2

NUMBER TO BE FACTORED ?0

corrected

# TINY BASIC IL

```

INTERPRETIVE LANGUAGE SUBROUTINES
EXP:  TST  E4  TEST FOR UNARY '-'
      DB  '*-' OR 2000
      ICALL TERM  INPUT TERM ON AESTK
      NEG  E1  NEGATE VALUE ON AESTK
      NOP

E0:  TST  E01  TEST FOR UNARY '*'
      DB  '*-' OR 2000
E01:  ICALL TERM  INPUT TERM ON AESTK
E1:  TST  E2  TEST FOR ADDITION
      DB  '*+' OR 2000
      CALL  TERM  GET SECOND TERM
      ADD  E1  INPUT SUM OF TERMS ON AESTK
      NOP  E1  LOOP AROUND FOR MORE

E2:  TST  E3  TEST FOR SUBTRACTION
      DB  '*-' OR 2000
      CALL  TERM  GET SECOND TERM
      SUB  E1  INPUT DIFFERENCE OF TERMS ON AESTK
      NOP  E1  LOOP AROUND FOR MORE

E3:  RTN  THIS CAN BE RECURSIVE

TERM: ICALL FACT  GET ONE FACTOR
      TST  I1  TEST FOR MULTIPLICATION
      DB  '**' OR 2000
      ICALL FACT  GET A FACTOR
      MPY  I0  INPUT THE PRODUCT ON AESTK
      NOP  I0  LOOP AROUND FOR MORE

I1  TST  I2  TEST FOR DIVISION
      DB  '*/' OR 2000
      ICALL FACT  GET THE DIVIDEND
      DIV  I0  INPUT DIVIDEND BY AESTK
      NOP  I0  LOOP FOR MORE

I2  RTN  RETURN TO CALLER

FACT: TSTV  F0  TEST FOR VARIABLE
      IND  GET INDEX OF THE VARIABLE
      RTN
F0:  TSTN  F1  TEST FOR NUMBER
      RTN
F1:  TST  F1  ERROR IF [I]S NOT A 'C'
      DB  '*C' OR 2000
      ICALL EXPR  THIS IS A RECURSIVE PROCESS

FE1: TST  FE1  EVERY 'C' HAS TO HAVE A ')'
      DB  '*C)' OR 2000
      RTN

RELOP: TST  R0  CHECK FOR '='
      DB  '*=' OR 2000
      LIT  0
      RTN

R0:  TST  R4  CHECK FOR '<'
      DB  '*<' OR 2000
      TST  R1  '*>' OR 2000
      DB  '*>' OR 2000
      LIT  2
      RTN

R1:  TST  R3  CHECK FOR '>'
      DB  '*>' OR 2000
      LIT  3
      RTN

R3:  LIT  1
      RTN

R4:  TST  R4  '*=' OR 2000
      DB  '*=' OR 2000
      TST  R5  '*<' OR 2000
      DB  '*<' OR 2000
      LIT  5
      RTN

R5:  TST  R6  '*=' OR 2000
      DB  '*=' OR 2000
      LIT  3
      RTN

R6:  LIT  4
      RTN

```

```

START POINT, TIME GRAN
)
)
) STATEMENT EXECUTION ROUTES IN IL (INTERPRETIVE LANGUAGE)
) THIS IS WRITTEN IN MACROS FOR THE INTEL INTEL 8080 AND
) SYSTEM USING INTEL'S ASSEMBLER
)
) CONTROL SECTION
)
START: INIT  INITIALIZE
ERRNT: NLIN  WRITE A CR-LF
CO:  GETLN  WRITE PROMPT AND GET A LINE
      TSTL  AEC  IF NO LINE VISIBLE DO EXECUTE IT
      INSR  LOOP FOR WHILE THE LINE
      IJMP  GO  LOOP FOR ANOTHER LINE
AEC:  AINIT  INITIALIZE FOR EXECUTION
)
) STATEMENT EXECUTION
)
S1:  TST  S1  CHECK FOR 'LET'
      DB  '*LE', '*' OR 2000
S11: TSTV  S1  ERROR IF NO VARIABLE!
      TST  S2  ERROR IF NO '='
      DB  '*=' OR 2000
      ICALL EXPR  INPUT EXPRESSION ON AESTK
      DONE  CHECK FOR CR LINE TERMINATOR
      STORE  INPUT VALUE OF EXPRESSION IN ITS CELL
      NXT

S2:  TST  S3  CHECK FOR 'GO'
      DB  '*G', 'O' OR 2000
      TST  S2  CHECK FOR 'GO10'
      DB  '*1', '0' OR 2000
      ICALL EXPR  GET THE LABEL
      DONE  CHECK FOR CR LINE TERMINATOR
      AFER  ADD A 'GO10' TO THE LABEL

S3:  TST  S2  CHECK FOR 'GOSUB', FAILURE IS AN ERROR!
      DB  '*SU', 'B' OR 2000
      ICALL EXPR  INPUT EXPRESSION ON AESTK
      DONE  CHECK FOR CR LINE TERMINATOR
      SAV  SAVE NEXT LINE NUMBER IN BASIC TEXT
      AFER  ADD A 'GOSUB' TO THE LABEL

S4:  TST  S6  CHECK FOR 'PRINT'
      DB  '*PRIN', 'T' OR 2000
S41: TST  S7  CHECK FOR '"' TO BEGIN A STRING
      DB  '*"' OR 2000
      SPC  PRINT THE DATA ENCLOSED IN QUOTES
      SPC  ' ' MEANS MORE TO COME
      HOP  SPACE TO NEXT ZONE
      HOP  GO BACK FOR MORE
      DONE  CHECK FOR CR LINE TERMINATOR
      NXT  CONTINUE NEXT LINE

S5:  TST  S9  CHECK FOR 'IF'
      DB  '*I', 'F' OR 2000
      ICALL EXPR  GET THE FIRST EXPRESSION
      ICALL RELOP  GET THE RELATIONAL OPERATOR
      ICALL EXPR  GET THE SECOND EXPRESSION
      SBA  CHECK FOR 'THEN'
      DB  '*THE', 'N' OR 2000
      CMPL  IF NOT TRUE CONTINUE NEXT LINE
      IJMP  SIM1  IF TRUE PROCESS THE REST OF THIS LINE

S6:  TST  S12  CHECK FOR 'INPUT'
      DB  '*INPU', 'T' OR 2000
      ICALL VAR  GET THE VARIABLE'S INDEX
      INJMP  GET THE NUMBER FROM THE TELETYPE
      STORE  INPUT THE VALUE OF THE VARIABLE IN ITS CELL
      TST  S11  ' ' MEANS MORE DATA
      DB  '* ' OR 2000
      DONE  CHECK FOR CR LINE TERMINATOR
      NXT  CONTINUE NEXT LINE

S7:  TST  S13  CHECK FOR 'RETURN'
      DB  '*RETRN', '*' OR 2000
      DONE  CHECK FOR CR LINE TERMINATOR
      NSTR  RETURN TO CALLER

S8:  TST  S14  CHECK FOR 'END'
      DB  '*EN', 'D' OR 2000
      FIN  GO BACK TO CONTROL MODE

S9:  TST  S15  CHECK FOR 'LIST'
      DB  '*LI', 'S', '*' OR 2000
      DONE  CHECK FOR CR LINE TERMINATOR
      LSI  TYPE OUT THE BASIC PROGRAM
      NXT  CONTINUE NEXT LINE

S10: TST  S16  CHECK FOR 'RUN'
      DB  '*RU', 'N', '*' OR 2000
      DONE  CHECK FOR CR LINE TERMINATOR
      NXT  CONTINUE NEXT LINE

S11: TST  S16  CHECK FOR 'CLEAR', FAILURE IS AN ERROR!
      DB  '*CLEA', 'R', '*' OR 2000
      IJMP  START  REINITIALIZE EVERYTHING!
)
)
) ****

```

Since the last issue came out, the IL code, macro definitions for each IL instruction, a subroutine address table for the assembly language routines that execute the IL functions, the assembly language code that executes the IL functions (all except the 16-bit arithmetic ones), and the IL processor have been punched on paper tape in source form.

HOP, TST, TSTN, and TSTL now do branches +32 relative to the current position counter. If the relative branch field has a zero in it, indicating a branch to "here", the IL processor prints out the syntax error message with the line number. The ERR instruction that was in the old IL code no longer exists.

IJMP and ICALL are used because the Intel 8080 assembler uses JMP and CALL as mnemonics for 8080 instructions. IJMP and ICALL are followed by one byte with an unsigned number from 0 to 255. This is added to START to do an indexed jump or call.

Bernard



**ANNOUNCING  
A NEW  
NEWSLETTER**

A XEROX NEWSLETTER FOR SERIOUS TINY BASIC IMPLEMENTORS. 3 ISSUES FOR \$3.00. CONTRIBUTIONS CHEERFULLY ACCEPTED. (SEE PAGE 30 FOR MORE DATA).

I WANT TO SUBSCRIBE TO THE TINY BASIC NEWSLETTER:  
 NAME: \_\_\_\_\_  
 ADDRESS: \_\_\_\_\_  
 CITY: \_\_\_\_\_ STATE: \_\_\_\_\_ ZIP: \_\_\_\_\_



# Buyer's Guide



## Or — It's nice, new and shiny, but is there anything inside? By Keith Britton

This is the Christmas of the Year of the People's Computer. MITS, this Spring, sold the first computer ever mass-produced for the home and/or hobby markets—the first production computer for people rather than for business or science. Before summer, rumors abounded of new, wonderful, and better machines soon to be released. All summer we waited breathlessly for them to appear. And waited and waited. By late Fall some of them have appeared, and most of the new entrants have at least shown a prototype. So now is a good time for those who want to buy a computer for Christmas to take a long, hard look at what is being offered (and what *appears* to be being offered).

Insiders in the computer world learned a long time ago that things aren't always what they seem. And it isn't just that products aren't always what the advertising says they are: sometimes there isn't any product at all. This can be deliberate. It has been known for a company to announce a new machine in glowing terms, although it did not exist, solely as a maneuver intended to inhibit potential purchasers from buying from a rival. Where this was a straight bait and switch ploy, deliberate intent is obvious. Sometimes the intent may be more honest although the effect is the same. Markets, by definition, are competitive and the timing of the announcement of a new product may be critical to its success. The electronics industry is peculiarly sensitive to this for several reasons. The crudest but perhaps most obvious is money. Development requires it. If the company does not have it, and you want the product, you must provide it—usually in the form of advance payment. Thus you put up money in advance, often get a good deal, and get to the head of the queue.

When the product arrives—if it arrives—you are not in a good position to complain if it isn't what you paid for. But without this money many small firms would not make it into production and many fine ideas could never be developed.

Fine ideas are not enough though, they have to be practical. There are lots of things which are really nice *ideas*. Viatron had one. Holographic memories driven by lasers are another. Some things work well in the laboratory but cannot be made in quantity. All of which adds up to the question: Will it materialize? The press release will give some indication—though the man who wrote it may not know either. If it sounds too good to be true—it probably isn't.

Companies tend to be optimistic about their production schedules - even experienced ones. Design does not take too long, but then the prototype doesn't work quite right and redesign blows a few more precious days. Manuals are written slowly and are *very* expensive if they have appreciable graphic content. Software (for any major program like a BASIC compiler) takes man/years rather than man/months. Then there is the problem of suppliers who don't meet delivery dates...

If you propose to buy from an established company which has produced and sold similar items before, the problems are probably well understood. Delivery will probably be only a little late and their promises will probably be as reliable as the promises they made in their advertising for other products. If you have ordered from a guy in a garage or a company new to the business, anticipate a learning curve and try to guess where they are on it.

Software is the worst problem for both the owner and the manufacturer of a computer kit. MITS committed themselves to \$180,000 of payments for software. Because of this, Altair computers cost a little more but Altair owners already have a superb BASIC to work with and a bigger and better one coming. Unfortunately MITS, nervous about the size of their commitment, upset people with a horrendous "Software Agreement" which they required all of their software user's to sign (a requirement now dropped).

This encouraged antagonistic attitude towards MITS and resulted in large scale pirating of their software particularly amongst the hobby clubs. These clubs exist for their members to help one another, but when they help one another to someone else's property, harm is done. The belief, amounting to ideology in some quarters, that all software should be free is markedly more common amongst those who want to use it than those who write it for a living. MITS will probably make enough to pay its software bill, but other companies will not have missed the point. Hobbyists can now expect more software buried in hardware or paid for as part of the price of the system, which may ultimately have the effect of pricing out the small innovator.

This learning curve is well shown by MITS' performance over the last year. The 8800 showed much evidence of hurried design and production. Their market was wildly underestimated and they had problems with suppliers. There were retrofits and fixes, even errata to the errata. But six months later they were showing the 680. The untidy wires were gone and they clearly intended to run their prototype for a couple of months to thoroughly debug it before they sold any. There weren't any important shortcuts in the design which traded quality for price, and it was well judged to be what the market wanted.

The battle between rival micro processors continues to heat up. INTEL's 8008 is passing into history but their 8080, as we expected, is becoming a big winner. It is now being second sourced and rivals are cheap enough to force prices way down. INTEL is selling new faster versions—expect that their competitors will offer pin for pin and instruction for instruction compatibility—plus extra goodies. Surprisingly, only MITS and a new entrant IMS seem to prefer the 8080. Most of the new kits are Motorola 6800 based.

The 6800 appeared to have a lot going for it. It ran on 5 volts simplifying power supply problems, had an excellent interrupt handling capability on chip and two accumulators plus an index register—widely felt to be better architecture than that on the 8080. Some of the early chips available to hobbyists in the Bay Area appeared to have timing problems. We didn't mention this in the June PCC since it was probably just a single defective batch. Other things have surfaced since, though. *Compare index* works like any other *compare*—but it turns out that while *branch equal* and *not equal* work okay, *branches* using the N and V flags don't. They only work on a comparison of the most significant bytes. AMI's Assembly Language Programming Manual makes this clear, but it remains a trap for the unwary. Another trap is the assumption that the *on chip interrupts* are effectively isolated from each other. Under some circumstances they aren't. Though there is a partial fix, by software, a hardware glitch remains.

The dark horse in the 8 bit race is MOS Technology. They have opted for one accumulator and two index registers. We haven't had input from users yet—so we don't know whether they have problems. The real problem for them is lack of software—though the 6800 is not much better in this respect.

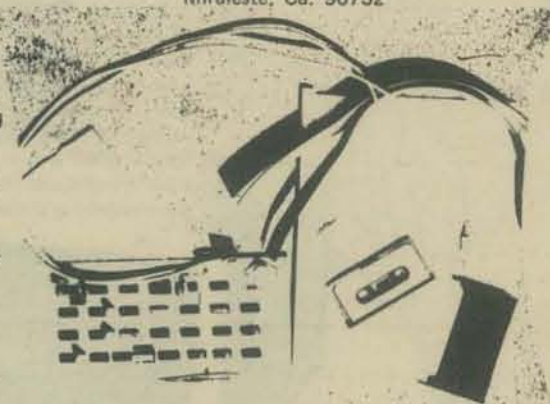
In the 16 bit field there are group buys afoot for the LSI-11 from DEC, but the real winner still looks like National's PACE, and it looks likely that there will be several PACE based systems to choose from. Bill Godbout's prototype has been running flawlessly since August—which is a good omen for the reliability of this chip. Here again, though, software availability may prove to be decisive factor in choosing a kit.



Everybody who has or wants a computer sooner or later arrives at the question: what do I use for mass storage? As soon as a program or data base grows beyond a few bytes, some storage medium is needed. Paper tape has some good storage features but fast readers are expensive and 10 characters per second is a pain.

The only medium which is reasonably cheap, fast, durable portable and commonly available is the voice grade Phillips cassette. There are, naturally, a variety of schemes for recording on it, many good, some lousy. Obviously, any one will do for your own system, since you record and play back on the same equipment. But if you wish to exchange tapes, all parties in the exchange must be in the same system. Because of this *Byte Magazine* promoted a meeting in Kansas City between various kit manufacturers and other interested parties to hammer out a standard.

The standard picked was 8 cycles of 2400 Hz for a 1 and 4 of 1200 Hz for a 0. This suited a 300 Baud transfer rate, the recording characteristics of cheap tape recorders and phone line transmission. It also had the advantage of being *very* cheap. We won't, of course, know for some months how widely this will be adopted, but if your primary interest lies in recording exchangeable cassettes, it seems advisable to wait and see. If you don't want to wait, the penalty for buying something else will not be severe since the projected cost for the exchange standard board is projected to be only about \$30 and you can always add it later. Which brings us back to the choices for immediate purchase. There are too many for us to cover them all, but there is one which is somewhat different from the rest, unlikely to be widely advertised but worthy of consideration - Don Tarbell's.



**Tarbell**  
144 Miraleste Drive #106  
Miraleste, Ca. 90732

This *will* record and read the proposed standard (with jumpers and a suitable software routine), but in its native mode it is *much* more powerful than the interfaces designed specifically for the new standard. Almost anything is satisfactory for loading programs, but if you plan to use files - you need this extra power.

Don Tarbell has had his own system running for a couple of years, using a recording scheme long popular in industry—called phase encoding. This records a clock signal and a data signal together onto the tape, which means that speed variations in the tape drive are not critical because both the clock and data vary at the same rate. This is good news for peoples with cheap tape drives. It also allows much denser recording than most of the alternative schemes. Don routinely uses 800 BPI (bits per inch) and has experimented recording at as much as 2200 BPI. Now this has two really desirable results as far as the home system is concerned. You get more data on a given tape, and you get more data per second—hence a higher transfer rate. How much? Well the proposed standard works out at around 160 BPI so with the recommended density of 800 BPI, a tape which would take half a minute to load at the proposed standard would take only six seconds with Tarbell's. This makes a lot of difference, and if experiment shows that your equipment is able to work with denser recordings, you can gain even more speed. The kit costs \$100 (\$150 assembled and tested). So it isn't cheap. We can attest, though, to its being of the highest quality and it comes with copious documentation. If you are sensitive about money, there are some savings with it; you need to buy a lot less tape

# Cromemco

One First Street  
Los Altos, Ca. 94022

Cromemco is offering a number of Altair compatible boards. An extender card and a wire wrap prototyping board (\$35 each) will be appreciated by hardware types, but of more general interest are their Cyclops camera/interface and their PROM board. The latter not only holds non-volatile PROM's but programs them as well!

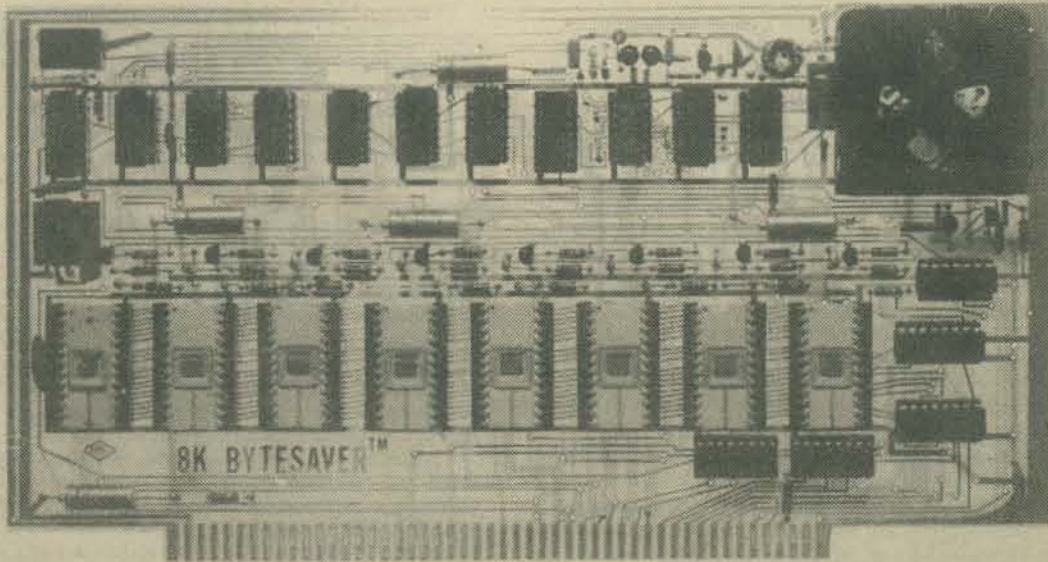
We have put together Cromemco's Image Sensing Module Kit (\$90), a single board which provides digital or analog signals from an image focussed on one of its ICs. The kit is fine - if one succeeds in building it correctly. The instructions are primitive and component placement has to be done by reference to a series of muddy and almost indecipherable photographs. Not for beginners!

The Cyclops, which was written up in *Popular Electronics* this spring, gives a 32 x 32 matrix picture. Use it for intrusion alarm, OCR, or as an eye for your robot.



It is not cheap though - kit \$180 for camera plus \$260 for an Altair controller kit, and the latter uses two of your precious slots.

\$195 - Kit Price



\$295 - Assembled

The Cromemco Bytesaver<sup>®</sup> Memory Module is a full speed 8K capacity ROM board with built in PROM programmer. The Bytesaver plugs directly into the Altair 8800 Computer.

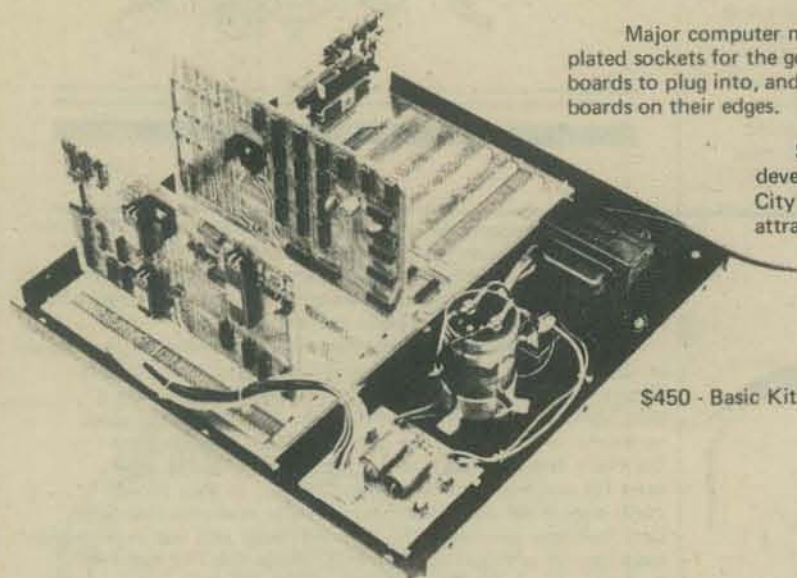
The Bytesaver provides a new convenience in program storage. Once a PROM is programmed the power may be turned off without affecting the contents of the PROM. The PROM's used in the Bytesaver may be erased with ultraviolet light, so that they may be used again and again.

The Bytesaver may be used with 2304 or 2308 ROM's or with 2704 or 2708 PROM's. The 2704 is a 512 byte PROM. The 2708 is a 1K byte PROM. Both the 2704 and the 2708 are high speed-devices (450 nanosecond access time) that allow the computer to run at full speed. A PROM in any of the eight sockets on the Bytesaver board may be programmed under software control. A protect switch, located in the upper left corner of the board, may be used to disable the PROM programmer to prevent accidental PROM programming.

# SWTP

219 W. Rhapsody  
San Antonio, Tx 78216

Kit Price - \$175



\$450 - Basic Kit

Major computer manufacturers learned to use gold plated sockets for the gold plated edge connectors of boards to plug into, and they learned to support the boards on their edges.

SWTP was one of the kit makers who (independently) developed the recording scheme selected by the Kansas City conference - so their cassette interface is particularly attractive.

SWTP has decided to do without a front panel, saving money and complexity, and go Hexadecimal rather than Octal. Octal is much easier to learn and to teach but many source programs are available in Hex. The lack of a front panel will prove a disadvantage to anyone serious about assembly programming, but would not bother someone running in BASIC. Unfortunately, we do not know of any BASIC likely to be available for the 6800 within the next few months. Lack of a front panel also means that nothing can be done with the machine without some I/O device.



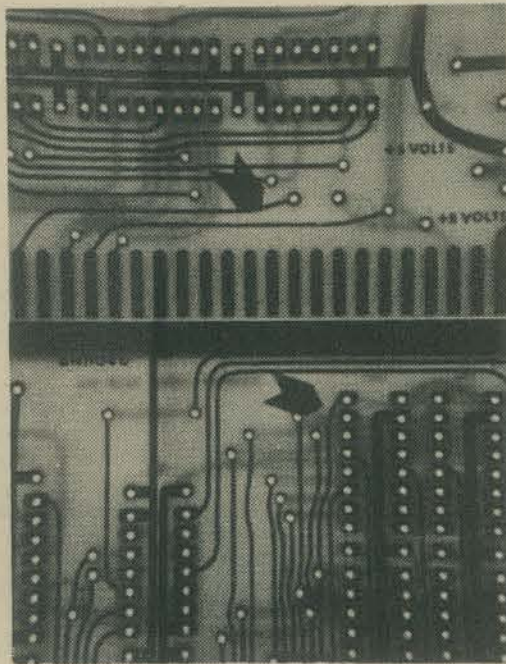
The design philosophy is clearly the same as that in the TVT II—a large mother board with other boards plugged in at right angles on Molex connectors, all boards having a notably low component density. The latter is fine. Really dense boards look very well but are more difficult to build, and a bitch to work on when something goes wrong. The former, Molex connectors, have proved adequate for the TV typewriter but we still don't like them. They are certainly cheaper than sockets (pennies against \$6 for the 50-pin connectors MITS uses) but they are not as strong mechanically. They have been known to stress the mother board enough to break traces, so if you intend to drop your computer or subject it to vibration, you may be in for trouble.

However SWTP TV Typewriter, which we reviewed in PCC, Vol 3 no. 5, fits directly against the computer so buying both solves the I/O problem. One nice thing - the 4K memory boards required for expansion are comparatively cheap at \$125.

# Godbout

Box 2355  
Oakland Airport, Ca. 94614

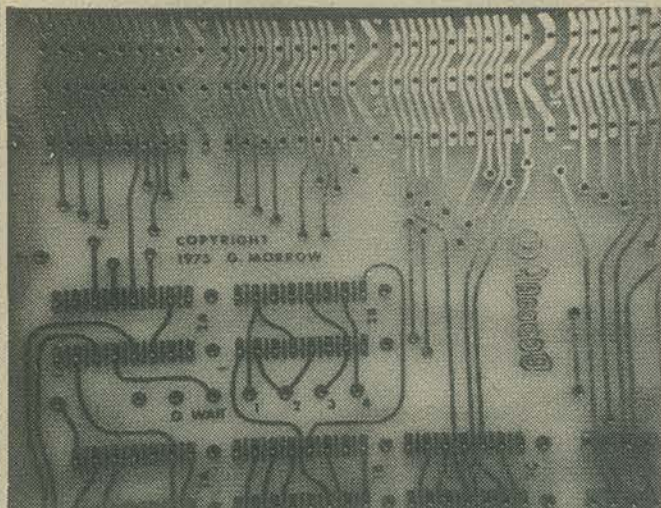
We have been to visit Bill Godbout several times over the last few months. Our original interest lay in his bona fides as a parts supplier and as the developer of the first PACE based kit. Since then, we have come to value him as a resource for technical advice and information. So it is, perhaps, somewhat heartless of us to publish a photo of defective Godbout parts (two printed circuit boards) but it does tell a story worth telling. Bill has been about in the industry since the days he helped IBM into computing. He has both definite ideas about quality and extensive experience maintaining it. He doesn't use PC houses which do not minutely inspect his boards, as the arrows (markers put on by the PC house) attest. Reproduction on newsprint may not be good enough to show the flaws, but they were sufficient for Bill to scrap the board. (He is giving his rejects to PCC for us to distribute to non-profit hands, which is how we got them.) We have seen a lot of boards sold to hobbyists which were far worse than those Bill scraps, so we got to talking with him and George Morrow about quality control and other things. We mentioned that we had heard people say that he buys reject parts and could hardly believe that it was true in view of the care we saw his staff taking to achieve quality. Instantly, he said that not only was it true, but he bought all the cosmetic rejects he could get his hands on. He then went on to explain to us what a cosmetic reject consists of. Many parts are ordered from a major manufacturer to be marked with the ordering company's part number on it. If the order is cancelled, the parts are cosmetic rejects; cosmetic because, they don't look right. If the proper number is smudged, put on upside down or in the wrong place, the part will also be a cosmetic reject. These parts are, of course, electrically perfect. Bill will buy them - cheap - and sell them - cheap. Which explains why his prices are currently amongst the lowest on the market but without compromise on quality. His 4K Altair boards, for instance, at \$109 have gold plated connectors, sockets for IC's and no less than five voltage regulators to spread the load. (Buy enough of them to get his normal discount - 100 at 20% and the price is around \$83 - attractive for a club!) Likewise his single memory chip prices - \$1.95 for one 2102, but at the 20% discount for 100 \$1.53 and less than \$1.37 at the 1000 quantity 30% discount.



We asked what speed the chips were tested at. He told us - and we don't doubt that his customers will be satisfied.

His PACE kit is currently held up because of late delivery of necessary parts. (Yes, he does buy from the same suppliers as other computer manufacturers.) He is not going to announce that it is available until he has enough parts to fill all orders promptly. He already has an awful lot of parts. We estimate he has three tons of transformers for the power supply! Noteworthy, he will have an export power supply for the machine too - variety of voltages and 50/60Hz.

Our discussion happened to touch on the copying of PC board designs which is being done by some individuals connected with clubs and by others commercially producing boards. This is one Bill Godbout and George Morrow have learned the hard way. They have some of the best, and therefore most desirable, designs in the business. After having some of them stolen, they are putting a copyright notice on *everything*. Not just production boards and prototype boards, but even on the scraps of paper containing design doodles! It is a pity the world is this way, but expect others to be similarly burnt and to follow suit.

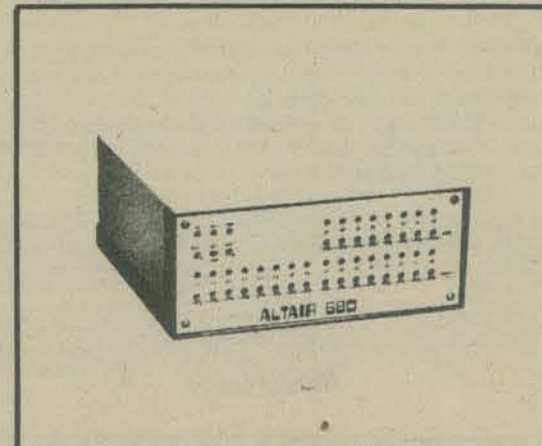


On the subject of PACE based kits, there are all sorts of rumours about new kits. Radio Shack is a likely candidate for a kit, according to our spies, and there are others. Bill Godbout's machine has been running for some time now and it seems likely that the advertising for the other kits will be targetted to sound a little better than his, since his is the only known entry so far in the 16 bit kit field. This could lead to a collision on price, goodies, or empty advertising claims designed to divert potential customers attention from machines which are in fact excellent. If you want a PACE kit - keep your money in your pocket until you are sure who is able to deliver what.



# MITS

6328 Linn, N.E.  
Albuquerque, N.M. 87108



MITS started the year - and the revolution - with the Altair 8800. It had its faults and MITS' competitors made the most of them. It is easy to design something to be superior to an existing product. You profit by the other fellows mistakes and, by the time you start designing, the state of the art has advanced and parts prices changed. MITS had to buy their 8080s at a time when Intel were selling the chip for \$359 in single quantities. MITS nevertheless sold a complete, though minimal machine for only \$80 more. MITS prices have carried the burden of pioneering - and of providing their users with BASIC. Tradition in the computer industry would now suggest that they rest on their laurels and make money selling their existing product while their competitors catch up. So what do MITS do? They listen, unconcerned, as their competitors claim improved designs and the then appear right out in front again with their 680 technically excellent and with a numbing \$289 price tag. Run harder fellas - the leader is still out there . . . and setting a blistering pace.



# Suppliers

We asked for input about suppliers - good and bad. It is pleasant to note that bad reports were few, however a clear top and clear bottom did emerge.

JAMES ELECTRONICS were mentioned often and only with an excellent rating. Address: P.O. Box 822, Belmont, CA 94002

Clear bottom was MicroMiniMart and, in view of their apparent reputation, we will not waste space on their address.



## Sphere

791 South 500 West  
Bountiful, Utah 84010

We saw their machine when they came to San Francisco for WESCON and have been interestedly watching their progress since. They were advertising hard and asking for advance payment, a tactic employed by small companies who need the capital or by rip off artists about to split for Brazil.

Just to set our minds at rest as to which they were... we paid them a visit. Sphere, it turns out, is at least hemispherical and becoming more solid and perfectly rounded every day.

They are admittedly, small as yet, but they are working hard and growing. One of the reasons for the hard work is that the workers own the company. This tends to lead to products a company can take pride in and to satisfied customers.

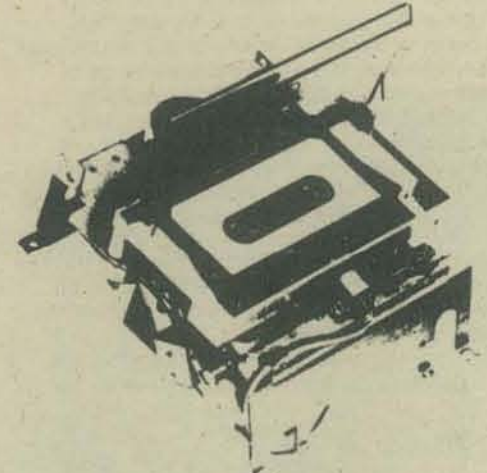
It also helps to promote honesty toward their customers. Sphere is about to publish the first issue of their own users group newsletter, and, like MITS' publication *Computer Notes*, it will be honest about mistakes and problems. It is clearly destined to be an important source of information regarding the 6800.

They are shipping both kits and assembled units though with some backlog on the latter. One good idea - the parts for their kits are being mailed pressed into a foam plastic sheet with the part identified on it. These kits don't get shipped with a part missing and the buyer has no difficulty identifying the next part he needs. Its obviously much more expensive than a plastic bag full of parts, but we hope others will consider using it. It's nice for the buyer.



## Phideck

ECONOMY COMPANY  
P.O. Box 25308  
Oklahoma City, Ok. 73125



For those who wish to build a miniature professional system using Phillips cassettes rather than going the El Cheapo cassette recorder route there is nothing to match the Phideck. It has four motor reliability and flexibility and an overall performance which has earned it reviews in the excellent to rave range. Would fit really well with a Tarbell interface - the best of everything. Note though that you have to build the controller for it; all you get is the drive and motors.

## Xebec

566 San Xavier Ave.  
Sunnyvale, Ca. 94086

We don't expect you to buy this one; at around 10 grand, it is not a hobby item! It is the shape of things to come though.



This bit is a replaceable ball point pen. When you write with it, sensors tell a minicomputer which direction you moved it, and the computer works out the number. Currently, it only works with about 14 characters, but this is still the bookkeeper's dream. A real advance over other "computer" pens because this one writes anywhere and on any paper.

Previously, there were three main styles of computer pens. The light pen picked up the sudden increase in brightness as the scan passed the pen. The computer knew where the pen was, but this was limited to writing on a CRT.

Tablet devices used the pressure of the pen to cause an electrical connection to a grid which the computer then looked at. The pen could only write on the grid and similarly "click" pens were only usable in a small area. These produced an ultrasonic click upon computer command. The computer then listened to the sensors, and from the delay, calculated the position of the pen. This new one from Xebec writes, as we said, anywhere. How does it work? They wouldn't tell us, but we guess that it has pressure transducers in the head.

Uses: Bookkeeping, obviously, but how about teaching a child to write? Instant feedback on forming letters...

## Jolt

MICROCOMPUTER ASSOC.  
111 Main St.  
Los Altos, Ca. 94022

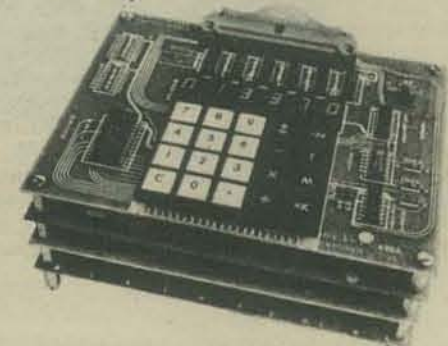
This is a new, heavily advertised kit. It uses the MOS Technology 6501 and consists of a series of boards, rather like the Mike II, with no front panel or cabinet. It was advertised as the world's cheapest computer at \$249 for the CPU board, but MITS for \$40 more, give you a whole computer. It does come with software in ROM, but there is not yet a high level language for this new CPU chip. The demonstration model worked fine at WESCON, but we have had no feed back from anyone who owns one.

## Mike

MARTIN RESEARCH  
3336 Commercial Av.  
Northbrook, Il. 60062

The Mike II is similar to the above, but the chip, INTEL's 8008, is thoroughly understood if much slower. Considerable software is available via *Micro-8* group. The boards remain an option worthy of consideration by anyone who has a lot of cheap but slow memory available.

It's upgradeable to 8080 with a CPU board change if you want to later.





# Processor Tech

2465 Fourth St.  
Berkeley, Ca. 94710

Kit Price \$160

Processor Technology continues to produce boards of unexcelled quality, with good documentation and very complete kits. Their two recent offerings are both worth comment—the (now famous) **Mother Board** and their **Video Display** module.

The **VDM** was designed by Lee Felsenstein after a lot of input from people who had ideas about what a *good* unit should have—and most of them were incorporated. Characters can be made black on white, white on black...the photo is its own story. (Joke currently in Berkeley—this is maybe the only advertising photo with a simulated TV. The screen is genuine photo, taken in a dark room, but the TV was added later....)

Processor Technology is a small company doing their best to produce fine products and succeeding at it. Everyone has problems though—and one has turned up in the first issue of their I/O board. (Write to them for the fix if you have one.) They stand by their products and deliver what they promise—good people to buy from.

The **Mother Board** is a massive chunk of 1/8" thick PC board which fits along the floor of an Altair. This holds lots of sockets and cleans up the interior but its principal advantage is the electrical improvements. Power and ground are both much better distributed (which has its points for those approaching the limits of their power supplies) and the bus is terminated.

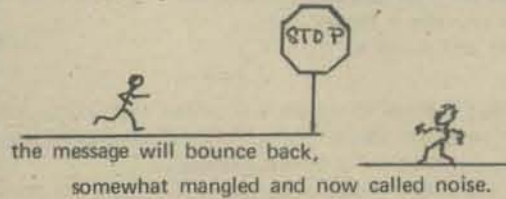
What is termination? Well it's like this:

We installed the PT Mother Board in our Altair. Here are the before and after pictures.

Altair Before

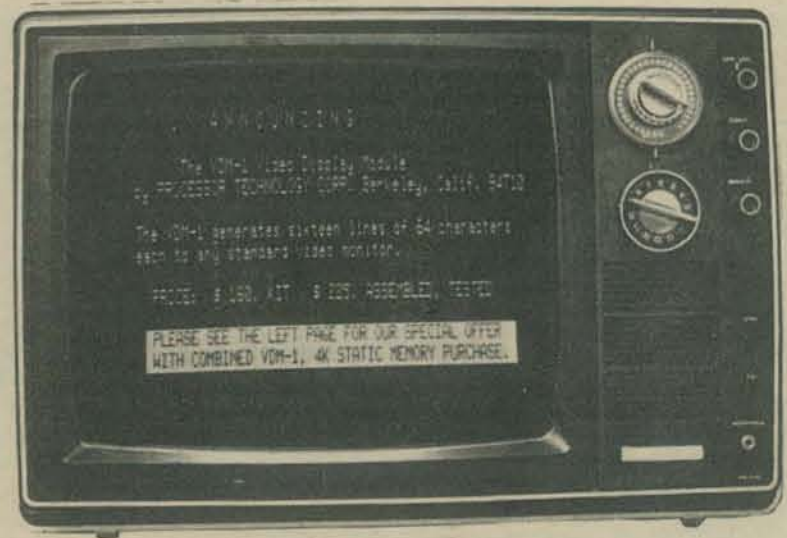


If you send a message down a line which comes to an end...



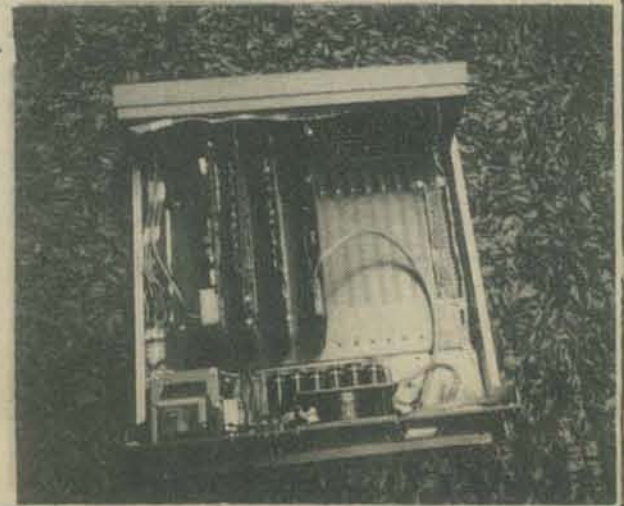
Processor Technology's termination is like having the line end at a place where there's a hole in the floor and a hole in the ceiling.

You send a message down it and *nothing* comes back.



Kit Price \$70

Altair After



# Logical Machine

887-A Mitten Rd.  
Burlingame, Ca. 94010

This one is expensive at \$40,000, but you get a machine which does not read or use software. It is interactively self-programming. Even children can use it and without learning a language like BASIC. You talk to it; it talks to you; eventually it finds out what you want it to do, and it does it. It starts with initial vocabulary of around 50 words, and the two of you build new words to add to it.



# IMSAI

1922 Republic Ave.  
San Leandro, Ca. 94577

IMSAI-30A  
Character Printer

CRI-1  
Cassette Recorder  
I/F

IMSAI-300A  
Line Printer

LIF  
Line Printer  
Interface

IMSAI-108  
50 Megabyte disk  
System

DISK-50  
50 Megabyte Disk  
and I/F

PIO 4-1  
1 Port Parallel  
I/O Board

PIO 4-4  
4 Port Parallel  
I/O Board

PIOM-1  
Parallel I/O  
Module

SIO 2-1  
Single Channel  
Serial I/O Board

SIO 2-2  
Dual channel  
Serial I/O Board

SIOM-1  
Serial I/O  
Module

SIOC  
Serial I/O  
Clock

PIC-8  
Priority Interrupt/  
Clock Board

PS-20  
Power Supply

A company which may be real competition for MITS, IMS Associates, Inc., is offering an alternative to the Altair 8800. The machine, named the IMSAI 8080, uses boards which are "plug in" interchangeable with Altair boards. The prices are similar with MITS, and delivery is to begin on December 15.

The IMSAI 8080 has a number of interesting features. The cabinet is a custom design with a built in card cage and an increased number of I/O socket locations on the back. A rack mount version is available.

The front panel plugs into a socket on the back plane which eliminates the wires found in the Altair 8800. The IMSAI 8080 can be used with the front panel removed. The switches are debounced and there are 8 LED's which are program controlled.

The power supply is advertised as 24 amps at 8 volts and 3 amps at each of +16 volts and -16 volts. The power supply is big and ugly and looks like it will deliver.

A priority interrupt and clock board handles 8 priority levels. Program control determines current permissible priority level. The clock interrupt has program selectable interrupts at .1, .2, 1, 2, 10, 20, 100, 200 and 1000 millisecond intervals. The board uses the INTEL 8214 chip and also provides a single bit output port with transistor driver and space for a 3" speaker.

For exotic applications IMS offers a three board set that allows multiple processors to share the same memory. A portion of the shared memory can be interlocked to allow software communication between programs in the different processors.

Shown in the picture above is the IMSAI ap-32 AlphaNumeric Printer. It prints with a dot matrix electrostatic head on special paper. Using a 5 x 7 dot matrix, 32 characters per line can be printed. The printing rate is two lines per second. The paper is 60mm (2.36 inch) wide so the characters are small. The printer drive requires 24 volts. The interface suggested by IMS is quite simple, or you can purchase a complete interface including software. The price of the printer is \$139 and the paper is \$3 for 3 rolls. This printer can be used with little trouble on anything which can supply 8 bit parallel output.

IMSAI 8080  
Microcomputer  
System

MPU-A  
Microprocessor  
Board

RAM 4-1  
1K static  
Memory Board

RAM 4-4  
4K static  
Memory Board\*

MM11-1  
1K static  
Memory Module

PROM 4-2  
2K EPROM Board

MM702-5  
512 work EPROM  
Memory Module

CP-A  
Control Panel

EXP-  
Extender Board

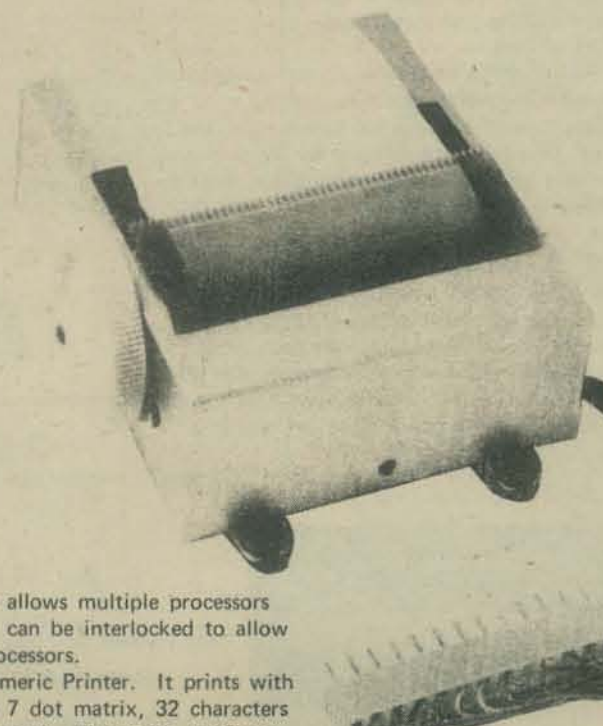
EXP-  
Extender Board

EXP-41  
EXP-81  
EXP-121  
EXP-161  
Expander Boards

FIF  
Floppy Disk Drive  
Interface

FDC-1  
Floppy Disk Drive

AP-32  
Alphanumeric Printer



## Gnat

8869 Balboa Ave., suite C  
San Diego, CA 92123

This machine has not yet been widely advertised. One of our spies told us about it though. Seems it was designed specifically as a machine for teaching about computers. Most computers have a front panel display but it is only intended to give information to a programmer, not a student. All computers have an instruction set - but it is intended to control the machine - not to show what is going on or to be easily understood by a human.

This computer has a front panel which displays all the internal activity and an instruction set which is primarily designed for people. Instructions are easy to understand, learn and are also representative of the types of instructions used by micro-computers. Price is about \$750 - not cheap but reasonable for a school and a bargain when compared to seminars and courses costing \$100+ per day.



## BlastMasters Inc.

P.O. Box 31  
Loma Mar, Ca. 94021

This (mysteriously) is the company through which our very own Bob Mullen sells his 'Mullen Computer Boards'. Available now - the cheapest extender board kit for the Altair that we know of at \$25. (Don't confuse extender with expander!). Available almost immediately - Relay/Opto-Isolator board for machine control for the Altair. Price \$65. (Build your own robot.) Available soon - A/D and D/A board for the Altair. Price not yet established \$90 ± 20.

## NEW - coming & secret

Last PCC we printed a photo of a new acoustic phone modem. This is almost in production (waiting for its case) and works just beautifully. Who is making it? That's a secret still. Find out all about it in one of the early spring issues of Popular Electronics. You will also have to wait for Popular Electronics to tell you all about the other hot new development. Named the TV Dazzler it is a color TV display for an Altair! It was demonstrated at the recent Computer Fair at the Lawrence Hall of Science in Berkeley and has been seen around Bay Area hobby clubs since, creating a minor sensation each time.

Seems like a subscription to P.E. is something one just has to have these days.....a nice Christmas gift at \$6.98 for a year. From: P.O. Box 2774, Boulder, CO 80302

# Analog to Digital Conversion

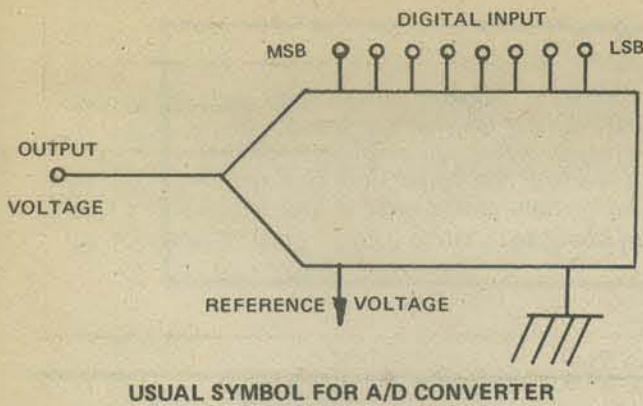
Many measuring devices are constructed to generate a voltage which is related to the property being measured. Example: a thermocouple generates a voltage which increases as the temperature difference between the hot and cold junctions increases. To use a voltage level (analog signal) in a digital computer it is necessary to convert to a discrete measurement (digital value). This process is called (guess) analog to digital conversion and is abbreviated A/D.

There are three basic approaches to A/D conversion. The first is to use the voltage to produce something we can count, such as pulse rate, frequency, or capacitor charging rate. Determining the digital value is then reduced to counting for a fixed period of time. The count is then the value. This method is slow and works best for near static signals. It is used in digital voltmeters.

The second method is by direct comparison with known voltages. This requires  $2^n - 1$  voltage levels for n-bit output. Each level needs comparison circuitry and logic to convert the results into usable form (binary or BCD). This method is very fast, but the amount of circuitry needed limits it to low precision (or very expensive) conversions.

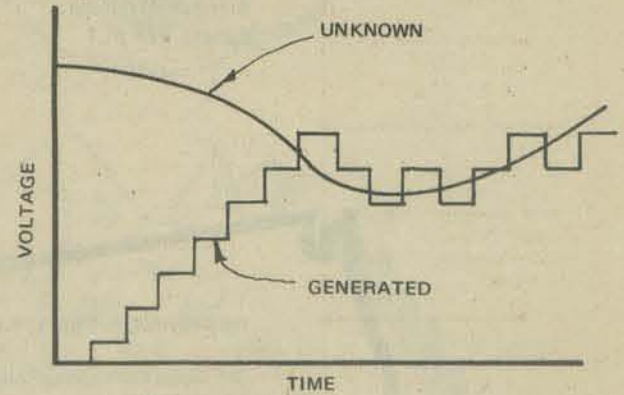
This third method is to compare the signal with a voltage which has been produced by some digital count. When the voltages compare within some limits, the digital count is used. This method can be made fast or cheap or in between, and is the method generally used for computer input.

The section which generates a voltage from a digital count is called a digital to analog converter (D/A) and is used in several different versions of A/D converters. A D/A converter uses a reference source, digital controlled switches, and a summing scheme. D/A converters can be purchased in small, inexpensive packages.



Comparison A/D converters use a D/A converter along with a digital counter, and a comparator. A ramp-type converter begins counting from zero and counts up until the generated voltage matches the unknown signal. This can require up to  $2^n - 1$  counts for n-bits, has a variable time, and is generally used with a sample and hold circuit.

A sample and hold circuit is used to get the voltage level of a changing signal and hold it for the time period needed for the A/D conversion to take place. A sample and hold circuit is basically a capacitor which is allowed to charge the voltage of the signal during sampling time and not allowed to discharge during the hold time.



When there is a converter for each signal, a tracking converter is ideal. If several signals are to be switched (multiplexed) to a single converter, a successive approximation converter is likely to be the best choice.

To convert a signal which goes plus and minus from ground (bipolar) it is necessary to use the sign bit. For a sign magnitude conversion the sign bit can be used to reverse the polarity of the reference voltages in the D/A section. For 2's complement conversion the reference voltage in the D/A section would use the most negative value in place of ground and work with the complement of the sign bit.

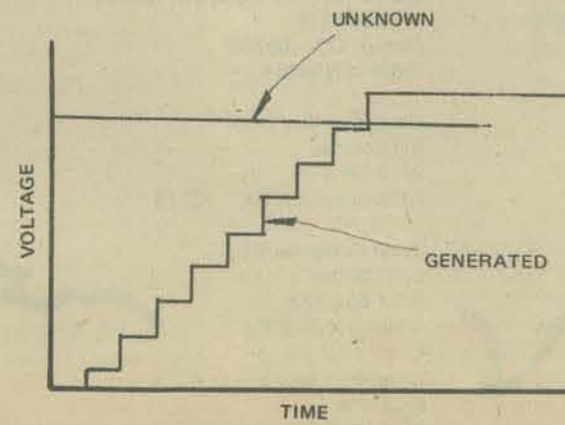
Signed Decimal	2's Complement	2's Complement with sign bit reversed
+3	011	111 Highest Voltage
+2	010	110
+1	001	101
0	000	100
-1	111	011
-2	110	010
-3	101	001
-4	100	000 Lowest Voltage

An alternative method is to switch in a minus reference when the sign bit is one.

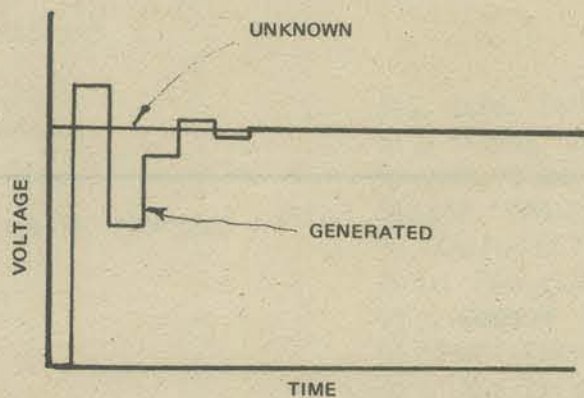
Signed Decimal	2's Complement	Voltage in D/A	
+3	011	3 volts added to ground	+3
+2	010	2	+2
+1	001	1	+1
0	000	0	0
-1	111	3 volts added to -4 volts	-1
-2	110	2	-2
-3	101	1	-3
-4	100	0	0

The voltage range of an A/D conversion could be made anything, but commercial units usually have ranges like: +5 to 0, +10 to 0, +5 to -5, and +2.5 to -2.5. The digital output is usually binary, but can be BCD or something else.

As with everything, it isn't just this simple; the whole process is loaded with pitfalls. The inaccuracy in an A/D converter must be less than the value of the least significant bit (LSB) or that bit has no meaning. In an 8 bit converter the value of the LSB is 1/256 of full scale or about .4%, in a 10 bit converter 1/1024 or about .1%, and in a 16 bit converter 1/65,536. While it is possible to design with parts not this precise, the ratios of component values must be, and must stay over the complete operating temperature range. If component values change with age, so will the accuracy of the conversion. Other areas of trouble are the ability of the converter to follow a changing signal, the linearity of the conversion, and what the converter does with an out of range signal.



With a slight increase in circuit complexity a successive approximation converter can be produced. A successive approximation counter turns on each bit one at a time, starting with the most significant bit (MSB). If the voltage generated is less than the signal it is left on, if the voltage is greater it is turned off. Conversion this way requires n-steps for n-bits. This method is also generally used with a sample and hold circuit.



Both the ramp and the successive approximation methods are generally begun with a start signal, at which time the unknown voltage is sampled, and the conversion is started. When the conversion is completed a 'done' signal is produced.

A third comparison method makes use of a counter which can count in two directions (up/down counter) and is called a tracking converter. A tracking converter can begin counting at a value other than zero, which reduces the maximum count to  $(2^n - 1)/2$ , but the real advantage is that after an initial time period the converter follows (tracks) the changes in the incoming signal. Unless the signal changes too rapidly, the digital output can be used without a start or done signal.



**CANADIAN CLUB**

Canadian Computer Club  
G. Pearen  
861 11th. St.  
Brandon, Manitoba  
Canada 47A 4L1

**CENTRAL UNITED STATES**

Chicago Area Microcomputer Users Group  
Bill Precht  
1102 S. Edison  
Lombard, IL 60148

Chicago Users Group  
Robert Swartz  
195 Ivy Lane  
Highland Park, IL 60035

Southeast Minnesota Amateur Computer Club  
2122 NW 17th Ave.  
Rochester, MN 55901

Universe Unlimited Users Group  
John E. Kabat  
11918 Forest Ave.  
Cleveland, OH 44120  
(216) 795-2565

Denver Amateur Computer Society  
PO Box 6338  
Denver, CO 80206  
(303) 458-5444

Oklahoma City Club  
Bill Cowden  
2412 SW 45th  
Oklahoma City, OK 73119

Texas Computer Club  
L. G. Walker  
Rt. 1 Box 272  
Aledo, TX 76008

Dallas-Fort Worth  
Bill Fuller  
2377 Dalworth Apt. 157  
Grand Prairie, TX 75050  
(214) 264-0111  
(214) 264-9017

El Paso Computer Group  
Jack O. Coats Jr.  
213 Argonaut Apt. 27  
El Paso, TX 79912

**Clubs and Pubs.**

**EASTERN UNITED STATES**

Amateur Computer Society  
Stephen B. Gray  
260 Noroton Ave.  
Darien, CT 06820

Boston Area Group  
John Vullo  
21 Sunset Ave.  
North Reading, MA 01864  
(617) 664-4271

New England Computer Club  
Byte Magazine  
Petersborough, NH 03458

Amateur Computer Group of New Jersey  
Union County Technical Institute  
1776 Raritan Rd.  
Scotch Plains, NJ 07076

Students Cybernetics Lab  
16 Linwood Ave.  
Buffalo, NY 14209

Pittsburgh Computer Club  
Fred Kitman  
400 Smithfield St.  
Pittsburgh, PA 15222  
(412) 391-3800  
(412) 276-6546

Southern Florida  
Roberto Denis  
11080 NW 39 St.  
Coral Springs, FL 33065

**WESTERN UNITED STATES**

San Diego Computing Society  
Gary Mitchell  
Box 35  
Chula Vista, CA 92012

Southern California Computer Society  
PO Box 987  
Sotuh Pasadena, CA 91030  
(213) 769-6596  
(213) 472-0380

UCLA Computer Club  
3514 Boelter Hall  
UCLA  
Los Angeles, CA 90024

HP-65 Users Group  
Richard J. Nelson  
2541 W. Camden Pl.  
Santa Ana, CA 90024

Homebrew Computer Club  
PO Box 626  
Mountain View, CA 94040

Bay Area Microprocessor Users Group  
4565 Black Ave.  
Pleasanton, CA 94566  
(meetings Thursday of each month at 7:00 pm  
at the Great Western Meeting Hall  
East 14th St and Davis in  
San Leandro, California)

*WOW! Look at all the clubs and publications that have sprung up to meet the needs of computer hobbyists. If we have not listed your club or publication, please write in and tell us about it.*

**PUBLICATIONS**

BYTE  
Peterborough, NH 03458 \$12.00/year \$1.50/copy

Computer Hobbyist  
Box 295  
Cary, NC 27511 \$6.00/year \$0.50/copy

Computer Notes  
MITS  
6328 Linn Ave NE  
Albuquerque, NM 87108

Creative Computing  
PO Box 789-M  
Morristown, NJ 07960 \$8.00/year

Electronotes  
Newsletter of the Musical Engineering Group  
203 Snyder Hill Rd.  
Ithaca NY 14850

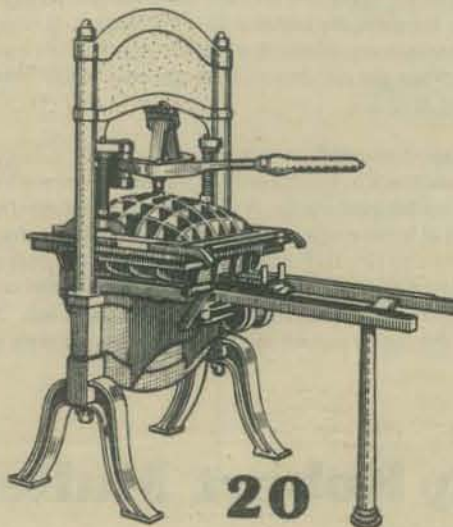
HP Educational Newsletter  
Hewlett Packard  
11000 Wolfe Rd.  
Cupertino, CA 95014

The Journal of Community Communications  
LGC Engineering  
1807 Delaware  
Berkeley, CA 94703 \$10.00/year \$1.00/copy

Micro-8 Newsletter  
Cabrillo Computer Center  
4350 Constellation Rd.  
Lompoc, CA 93436 \$6.00/6 issues

Popular Electronics  
PO Box 2774  
Boulder, CO 80302 \$6.98/year (12 issues)

Radio Electronics  
Boulder, CO 80302 \$8.75/year \$0.75/copy

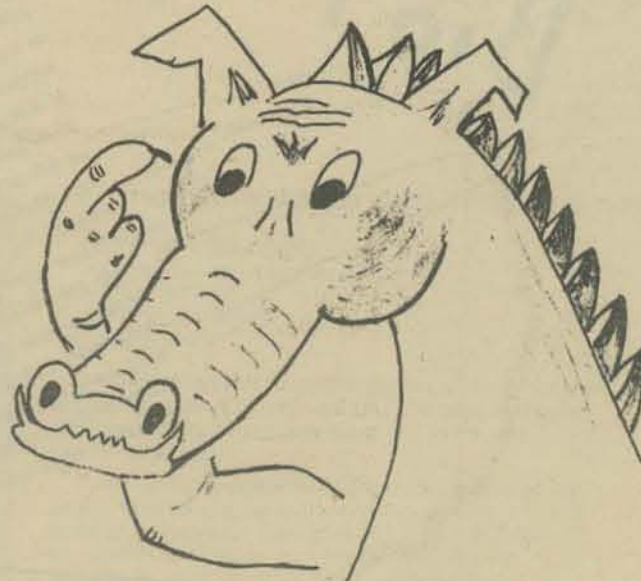


DEAR PCC,

I WANT YOU TO KNOW HOW MUCH I ENJOY YOUR MAGAZINE. I'VE REALLY LEARNED A LOT FROM IT. SINCE YOU SEEM TO LIKE PUZZLES SO MUCH, I MADE A CROSSWORD PUZZLE FOR ALL YOU DRAGONS. WATCH OUT, THO, IT'S A HARD ONE.

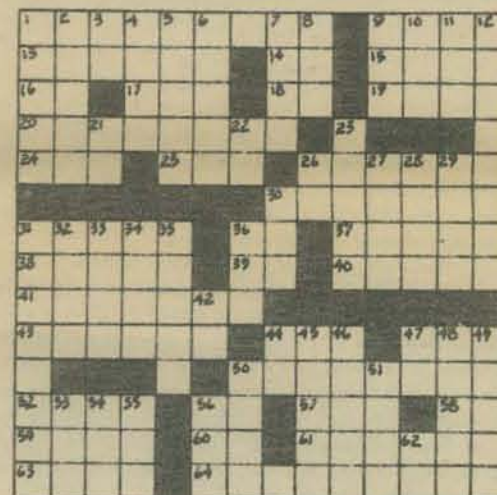
RON SANTORE  
1957 HUASNA DR.  
SAN LUIS OBISPO  
CALIF. 93401

ANSWERS ON PAGE 31



ACROSS

- 1 A DEVICE USED TO CONNECT A COMPUTER TO A PERIPHERAL.
- 9 A LOUD SMACKING KISS.
- 13 A NON-ALPHABETIC SYMBOL.
- 14 IN THE LOCATION OF.
- 15 A TOOL HAVING A HEAVY BLADE AT RIGHT ANGLES TO THE HANDLE.
- 16 ATOMIC SYMBOL FOR THALLIUM.
- 17 ABBR: BOY SCOUTS OF AMERICA.
- 18 ABBR: USED TO DENOTE A TIME CONSTANT.
- 19 ABBR: AMERICAN NATIONAL STANDARDS INSTITUTE.
- 20 FLEXIBLE OR STRETCHABLE MATERIALS.
- 24 ABBR: LARGE SCALE INTEGRATION.
- 25 SMART, CRAFTY, FOXY.
- 26 A CHARACTER FROM THE EAST, PORTRAYED BY JOHNNY CARSON.
- 30 TO REQUEST EARNESTLY, OR BEG FOR.
- 31 A SKEWER WITH MEAT & VEGETABLES.
- 36 ABBR: EAST INDIES.
- 37 A GIRLS NAME.
- 38 MAD.
- 39 ABBR: NOT.
- 40 A SUFFIX INDICATING CELLS; AS IN "LYMPHO\_\_\_\_\_" (PLURAL).
- 41 IMPAIRED GROWTH.
- 43 EDITOR OF THE MICRO-8 NEWSLETTER.
- 44 ABBR: RANDOM ACCESS MEMORY.
- 47 ABBR: IN AVIATION, "ABOVE MEAN SEA LEVEL" ALTITUDE.
- 50 EMPEROR OF JAPAN.
- 52 ABBR: TRANSISTOR-TRANSISTOR LOGIC (PLURAL).
- 56 TOWARD.
- 57 IN LYBIA, A MEASURE OF DISTANCE OR LENGTH.
- 58 A PREFIX INDICATING EGG OR CELL.
- 59 PLACED OR SET UP A BALL.
- 60 ABBR: CITIZENS BAND RADIO.
- 61 THE LOBE OF THE EXTERNAL EAR.
- 63 FINISHES.
- 64 PERSONS WHO USE COMPUTERS AS A PASTIME OR SPARE TIME ACTIVITY.



RON SANTORE

DOWN

- 1 A CHIP COMPANY.
- 2 IN MATHEMATICS: NUMBER SETS CONTAINING NO MEMBERS; ZEROS
- 3 THE ATOMIC SYMBOL FOR THULIUM.
- 4 FADES BACK OR RECESSES; AS THE TIDE.
- 5 STOPS WORKING.
- 6 FRAGILE.
- 7 AUTOS.
- 8 AND SO FORTH. (ABBR.)
- 9 ABBR: BACHELOR OF FINE ARTS.
- 10 A VASE FOR ASHES.
- 11 ABBR: DISTRESS SIGNAL.
- 12 DEVICES USED FOR OPENING OR CLOSING ELECTRICAL CIRCUITS.
- 21 A THREE TOED SLOTH.
- 22 A MAN'S NAME. (SHORTENED OR INFORMAL)
- 23 A COMPUTER LANGUAGE.
- 26 ABBR: KINETIC ENERGY.
- 27 DEPEND.
- 28 ~~THE~~ COMMERCIAL CREAM FOR HAIR REMOVAL.
- 29 PIMPLES.
- 30 IN BINARI; A 1 OR 0.

- 31 A TAPE PACKAGE FOR SOFTWARE OR MUSIC.
- 32 A PREFIX MEANING AGAINST.
- 33 ABBR: BEGUN.
- 34 ABBR: ORANGE.
- 35 COMPUTER WORDS.
- 36 IN PROGRAMMING, A STATEMENT MEANING FINISH.
- 42 A SUFFIX WHICH CHANGES ~~VERBS~~ VERBS INTO NOUNS.
- 44 ABBR: RHODE ISLAND.
- 45 A UNIT OF DRY MEASURE OF SEVERAL COUNTRIES OF THE NEAR EAST. (USUALLY 5.6 BUSHELS)
- 46 A TYPE OF TROPICAL, MARINE EEL.
- 47 ABBR: MICHIGAN
- 48 ERMINE
- 49 PROGRAMMING CIRCLES.
- 50 BUM.
- 51 "\_\_\_\_ RUD": AN ANCIENT NAME; ALSO A RIVER OF AFGHANISTAN & IRAN.
- 53 DECA INDICATES THIS NUMERIC BASE.
- 54 ABBR: LIGHT EMITTING DIODES.
- 55 ABBR: STUDENTS FOR A DEMOCRATIC SOCIETY.
- 56 ABBR: THE COMPUTER HOBBYIST.
- 62 ABBR: "LOCUS SIGILLI" THE PLACE OF THE SEAL.



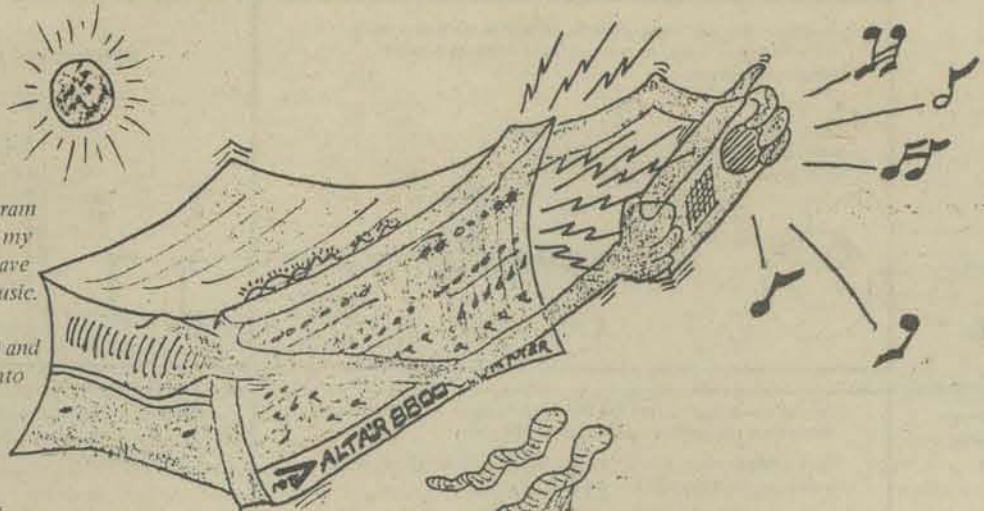
MINUET IN G MAJOR

by Anna Magdalena Bach

I loaded Stephen Dompier's Music Program which appeared in Volume 3 No. 5 issue into my father's Altair 8800. It worked great, but I have one suggestion for those who like Classical music. Here's some liberated Bach.

To use this program, insert address 040 and 000 into steps 001 and 002, and insert 060 into step 012 of Steve Dompier's program.

Cynthia Beyer age 12  
4145 Chestnut  
Long Beach, CA 90807



BUILD A HOUSE FOR YOUR MUSICAL ALTAIR

Address	Data	Address	Data	Address	Data	Address	Data
040	114	113	071	190	055	263	071
041	114	114	071	191	050	264	063
042	071	115	042	192	063	265	055
043	063	116	042	193	063	266	053
044	055	117	053	194	002	267	053
045	053			195	063		
046	114	120	046	196	055	270	055
047	114	121	042	197	050	271	055
		122	036			272	063
050	071	123	034	200	046	273	063
051	071	124	034	201	042	274	055
052	002	125	071	202	036	275	046
053	071	126	071	203	034	276	071
054	071	127	002	204	034	277	071
055	042			205	036		
056	042	130	071	206	036	280	075
057	053	131	071	207	042	281	075
		132	053			282	071
060	046	133	053	210	042	283	071
061	042	134	046	211	036	284	071
062	036	135	053	212	036	285	071
063	034	136	055	213	063	286	071
064	034	137	063	214	063	287	071
065	071			215	050		
066	071	140	055	216	050	290	377
067	002	141	055	217	046		
		142	053				
070	071	143	055	220	046		
071	071	144	063	221	046		
072	053	145	071	222	046		
073	046	146	063	223	046		
074	053	147	063	224	046		
075	055	150	055	225	002		
076	063	151	063	226	046		
077	055	152	071	227	046		
		153	075				
080	055	154	071	230	071		
081	053	155	071	231	036		
082	055	156	071	232	071		
083	063	157	071	233	071		
084	071			234	002		
085	075			235	042		
086	071	160	071	236	042		
087	063	161	071	237	071		
		162	002				
090	055	163	026	240	036		
091	071	164	026	241	071		
092	055	165	034	242	071		
093	055	166	031	243	002		
094	063	167	026	244	046		
095	063			245	046		
096	063	170	034	246	053		
097	063	171	002	247	053		
		172	031				
		173	031				
100	114	174	046	250	055		
101	114	175	042	251	055		
102	071	176	036	252	063		
103	063	177	046	253	071		
104	055			254	036		
105	053	180	002	255	071		
106	114	181	034	256	063		
107	114	182	034	257	063		
		183	042				
110	071	184	036	260	114		
111	071	185	046	261	105		
112	002	186	050	262	075		
		187	050				

HOW TO BUILD A HOUSE SIMPLY--FOR 1/3 COST

BY WILLIAM ZINK, P.H.D.  
JAY PUBLICATIONS  
726 STATE ST.  
SANTA BARBARA, CA.  
PRICE:\$5.50

When a research physicist decides to build his own house and write about it, you might expect a thesis level treatise mathematically justifying every step. Not so with this book.

Dr. Zink's purpose is to show the inexperienced person how he can build his own house with a minimum of professional subcontracting. His emphasis is on comfortable shelter and ease of construction. While he suggests that his house lacks "esthetic" quality, those who follow the "form follows function" school might disagree with him. His use of board-and-batten siding, which permits use of green lumber and no paint, is visually satisfactory to this old soldier. And, as the saying goes, you can laugh all the way to the bank.

Zink's approach to the question of what to do about the zealous amateur's trying for perfection in every detail was: "During the course of building, I adopted the attitude that I would never exceed that which was adequate."

In sum, with this book, and a hammer and saw, you are on your way.

OLD SOLDIER



# FORTMAN

Lee Schneider  
Todd Voros

As you know, our hero, Fortman, was born in a small system in New York, to normal, hard-working second-generation parents . . .

Look dear; he's learned his very first word!

LDA \*-1

carriage control tape

Even as a child, he stood out in a crowd . . .

Transfer Bus Stop

sigh

11 OF 100

But soon, it became apparent to the world that he had powers above and beyond that of the common compiler, impressing everyone with his speed, his mathematical ability, and his scientific powers!

Disguised as a mild mannered disk file, he keeps an ever vigilant watch on software in many places, always ready to spring forth to fight for Truth, Justice and the Algorithmic way!

Faster than a speeding Do Loop!  
More powerful than a Two-Pass Assembler!  
Able to leap tall matrices in a single Dimension!

It's a PC Board!  
It's a core panel!

No, it's **F** **FORTMAN**

As our story opens, we find our hero living in a resident-core district of a large metropolis called 360 City.

As he finishes his usual breakfast (a bowl of Parity Chex and a glass of Card Punch) he hears his hot-line ring . . .

munch munch

CTRL G!!

Quickly swallowing the last byte of Chex, he rushes to the line, and finds himself talking with the head of the city: the 360 Monitor!

Good thing I studied BCD in foreign language school; I can hardly understand this dialect! Wish he spoke ASCII like most monitors I know!

F-man, we need your help! Some villain has filled up all the Sysout files with meaningless output! The little town of Buffer is almost overrun!

A master of disguises, he hides himself as a seemingly harmless parity bit, and speeds to the nearest NAND gate, where he catches a high speed Data Transfer Bus, heading for the endangered town of Buffer. As he travels, he thinks to himself -

Meaningless output, he says! Sounds like my old arch-enemy, Cornelius Cobol! I must be on my guard; he's sneaky!

BUS STOP

Meanwhile, in a deserted index register on the other side of town . . .

At last! I've done it! With my new invention the Cobol Output File Compressing Machine, I'll squeeze all the bits out of these files and dump them on Buffer!

10110 Bit Bucket

C

Soon, the entire town of Buffer will be engulfed with my output! Nobody else will be able to get in! When that happens, Buffer will be mine!

This secret line lets me smuggle these bits through the data port without going through the gates!

hee-hee-hee

To town

Once I have this area under my control, I'll have an unlimited supply of files to feed my compressor! All of 360 City will be under my control!

CORNELIUS COBOL, will be master of the system!

Today, Buffer!  
Tomorrow, the world! But first, some lunch . . .

C

Biz-Ad Burgers

As Cornelius is busily munching his Biz-Ad Burger (and spitting error messages on the floor), Fortman arrives at the Output Queue, where he is backed up by the crowds -

hmmmm . . . I've never seen such a crowd . . . but I must get through!

NEXT!

Have your high priority output card ready

101011001

Q

Not wishing to reveal his identity, the quick-thinking F-Man slips into a nearby unused storage area, where, concealed behind the counter, he changes his disguise to that of a Priority Interrupt. This allows him to swiftly bypass the queue . . .

All right, folks! Move aside!

Thank you officer!

0100

Q

I

Not bothering to take time to change disguises, he quickly finds a city map . . . his great powers allowing him to memorize it instantly!

Map of Buffer

Good thing I took that Memory-Map course . . .

10 I 10110 011

Rapidly compiling this data, Fortman begins to formulate a plan of action . . .

If Cornelius Cobol is in town, he would most likely be underground! Means I'll have to check all the negative voltage supplies . . .

Voltage Ramp to Upper Level

He heads for the low-order part of town, where a number of known hangouts for dropped bits, invalid op codes, and other unsavory characters are located . . . when suddenly, he spots an acquaintance!

Well, if it isn't my old helper, little Billy Basic, boy reporter for the 360 City Daily Program! I'll bet he has some inside information I could use!

1 I 01 01 B 01

Slowly, so as not to arouse the suspicions of the surrounding riffraff, Fortman edges closer until he is right behind Billy Basic, and whispers the secret signal . . .

sysgen!

Golly, F-Man, I've been wondering if you would get here!

Can't talk here . . . Go To 999!

110 I B 1001

The pair split up, in order that they may travel separately to avoid attracting any undue attention . . .

They meet soon thereafter in the notorious 999, a two-bit hangout on the low-order side of town . . . a collection point for all sorts of dropped-out bits, degenerate functions, and useless redundancies . . . owned by the semi-retired gangster Big Mho!

When he first came to 360 City, Big Mho was a bridge worker for military specifications, but this work soon exceeded his limited tolerance, so he dropped his current project (which severely reduced his power rating), and went into business on his own . . . in an old, abandoned bus bar, where he kept busy hiding criminal elements, fencing stolen joules and helping draft resistors sneak out of town . . .

On the way, Fortman stops briefly and changes his disguise, so that he now resembles a dropped parity bit . . .

Allright, Billy, we can talk here! What have you got that might help?

Well last time I was in the print room, I looked over a copy of the next issue of the System Status Report, and it said there that Big Mho was suspected of smuggling large number of files into town illegally, without paying his Input tax!

I B



# LETTERS



## H2 for the Home Computer

My son John and I have built an Altair 8800, and I thought that I would share our experiences with you and your readers. John is thirteen and actually did most of the construction himself. This was John's first electronic construction job, and in spite of the complexity of the system, it was relatively easy for him to complete. This was true largely because of an excellent assembly manual put out by MITS.

The kit we bought included 8,700 words of RAM, and I/O board to interface a terminal, and an audio-cassette interface, plus an assembler and a BASIC compiler, all for \$1,160. It took us about forty hours to build the computer, partly because we were cautious about the construction, checking and rechecking each component before installing it.

Imagine how depressed we were when after all that care it didn't work when we plugged it in. I spent some time troubleshooting it to no avail. I'm not much good at troubleshooting, and gave up. I finally sent the machine back to MITS. It took their technician only one hour to find three defective IC's. When they were replaced, the machine worked, so John and I hadn't made any mistakes. Incidentally, MITS was very nice about this. I had to pay shipping charges but wasn't charged a nickel for the troubleshooting and repair. Their technicians also were very helpful on the telephone when I was trying to do my own troubleshooting.

So far I have only two criticisms of MITS and Altair. First, their assembly manual is excellent for construction but contains almost nothing to help the builder with troubleshooting. The second criticism I have is that it isn't possible to display the accumulator contents directly. If I want to see the accumulator contents, I need to store them in memory and then address that memory location.

We haven't yet gotten our BASIC compiler and are anxiously awaiting it. When we do, we plan to try out some Huntington simulations (how about that for slipping in a commercial) and some of those swift PCC games.

By the way, we tried Steven Dompier's music program (see PCC, May, 1975) and it works. It worked really nicely. I'm coding "The Entertainer" (theme song of "The Sting") for fun. This is interesting, but what sore fingers I get from the toggle switches.

I'm already convinced that the Altair 8800 and other machines like it are the wave of the future in home and classroom computers.

When we have had some time to try BASIC, we'll write to you again.

Ludwig Braun  
Director, Huntington Two  
College of Engineering and Applied Sciences  
State University of New York  
Stony Brook, NY 11794

## Songs & Poems

I am looking for poems and songs about computers. If anyone knows of any I would appreciate hearing from you.

D. Van Tassel  
Computer Center  
University of California  
Santa Cruz, CA 95064

## Lookin' for Stuff

I've recently learned of your organization and I'm very interested in what you're doing. I'm an elementary school teacher who knows relatively little about computers but still remains fascinated about the process of teaching/learning about computers in the elementary school.

Last year, for the first time, I spent about two weeks with my students exploring the different uses and functions of computer systems. While we learned a great deal, there remained a definite lack of adequate elementary school level (4th-6th grade) materials.

At this point I'm searching for materials and equipment for next year that'll allow my students to get involved in actual computing and, if possible, building a crude computer system. Do you know of any low-cost materials (books, simulated computers, etc.) and/or equipment that would fit my needs? If so, I'd appreciate hearing from you.

Frank S. Porter  
3135 "I" Street  
Sacramento, CA. 95816



## Whole Earth Club

The Whole Earth Club operates within the framework of the Orange County (NY) 4H clubs. We have about 20 members ranging in age from 8 to 16. They have a wide range of interests, but we tend to concentrate on technologies like electronics, photography, rocketry, computers and programming.

We have several people who are very interested in programming. Two 15 year olds have had considerable BASIC experience through courses at our Community College nearby. We are trying to find access to a BASIC (or APL) system, timeshare or locally, for our project on Computer Fundamentals and Programming later this fall. We are a fairly new group. We have no equipment of our own, but have healthy desires/fantasies of acquiring our own small system in the future. I personally own a model 33 KSR and have homebrewed a decent modem [schematic available for those interested], but have no access to a system except at painful long distance rates.

My wife Chris and I both work for the Snow White Computer Company here in Lower New York. She has been designing hardware to interface computers to manufacturing tools, and has also done some work on the associated system software. I have been writing and qualifying software for running measuring tools for making IC's and stuff, although I used to think I was mainly an electronics person. Anyhow, it seems like after a day of The Megabyte System JCL Sysabend Dump Trip it sure would be nice to come home to a warm, friendly little mini, the fireside, and a few kids working their way through the FOR/NEXT statement.

Getting back to reality, we really need teaching materials, or information on available materials, suitable for introducing program to, say, 10 to 15 year old people. Also, ideas on initial introduction to computing (joys thereof) to people of zero background would be appreciated. I look forward to developing materials and techniques as we go along, I just want to steal all I can to get off the ground.

About the Hardware Computer Hobbyist angle: I have strong interests in this area, and a lot of digital design/construction background. I am particularly interested in applying the Motorola 6800/MOS technology 2501 CPU's to a low-cost mini. I would like to hear of any work on a BASIC interpreter for this system. Later on I intend to do some work on interfacing to the outside sensor/actuator world by drawing on my wifes experience plus experimentation. Anyone interested in that subject?

We will keep you posted on our Club Project progress (and probably our problems) as we go along. You interest is appreciated. Seeing an issue of PCC really was an initial boost to us, just seeing that we weren't alone in our interests and ambitions.

Terry King  
The Whole Earth Club  
Orange County 4H Club  
P. O. Box 593  
Circleville, NY 10919

## Canadian Club

All persons interested in the formation of a Canadian amateur computer club are asked to write to the address below. The objects of the club would be the creation of standards for digital data exchange and the exchange of programs, hardware and application ideas etc.

G. Pearen  
861 - 11th Street  
Brandon, Manitoba  
R7A 4L1

## Pittsburgh Club

Just a quick note to let you know that hobby computing and computers are alive and well in Pittsburgh. As you can see from the logo the name of the organization is the Pittsburgh Area Computer Club. We had our first meeting on October 8, 1975 and even though the weather was inclement we had 15 people in attendance. In addition there is definite interest from at least that many more! Not bad for a beginning.

The name of our acting President is Eric Liber and the name of the Secretary/Treasurer is Fred Kitman. We can be reached at the address at the Bottom of this letter or at the following telephone numbers:

Day: 412-391-3800  
Nite: 412-276-6546

The members of the club have a total of 5 computers with the following breakdown:

Altairs	2
PDP/11	1
Wang	1
8008	1

(Not a bad start!)

We would appreciate the inclusion of our organization's name in your publication to inform other hobbyists in this area of our existence.

Eric Liber Fred Kitman  
Pittsburgh Area Computer Club  
400 Smithfield St.  
Pittsburgh, PA. 15222

Please include our name and address in your next issue of PCC. We are located in the Boston area and would like to increase our membership.

Our equipment includes an Altair computer with over 12K memory, ACR, SIOA, SIOC, PIO, VLCT, computer terminal, ASR 33 Teletype, TV Typewriter, Mark 8 computer and others.

John P. Vullo, President  
Alcove Computer Club  
230 Main St. RTE 28  
North Reading, MA 01864

## Computers & Society

Kings Park has been using its PDP 8E for teaching programming and computer math, and running various classroom simulations for the past four years. I am now trying to expand our computer science curriculum to include a "Computers and Society" type course for average students without any special math or computer background.

Although one of the goals of such a course would be to expose these non-math oriented students to the computer's capabilities, and to simple interactive programming techniques, a goal of at least equal importance would be the study of the social issues of computer usage and our increased reliance upon EDP.

There is no doubt that we are rapidly becoming a computerized civilization; but do we have the social sophistication to handle this new technocracy? Indeed, how many of our students, or their parents for that matter, are aware of how ubiquitous computer really are, and how dependent our business and science communities have become on these mystical black boxes. I believe that as computer science teachers we have a responsibility to raise and discuss such questions, to at least begin to stir a sense of social awareness.

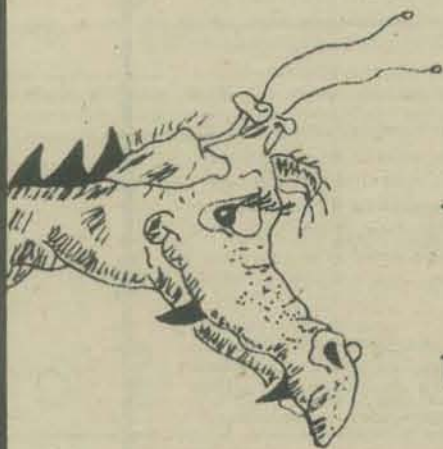
I would be most indebted to you (or to any of your readers) for any assistance you could offer:

- (1) Are you aware of any other high schools, or colleges, which have already developed such curriculum?
- (2) Could you recommend any references which I could use for my own edification in this area? (Yes, start with "Computer Lib/Dream Machines by Ted Nelson - see PCC Bookstore, page 31.)
- (3) Could you suggest any publications which would be suitable as texts or supplementary readings for my students? (*Blush... try PCC*)
- (4) Are there any areas which you think should be especially emphasized in such a program?

I have read John Kemeny's book *Man and the Computer* (Scribners, 1972). I have been favorably impressed with Tom and Fae Logsdon's textbook and workbook "The Computers in Our Society" (Anaheim Publishing, 1975). I have also read Grace Hertlein's article "Computer and Society" in the July, 1975 issue of "Computers and People", but although her outline mentions many interesting topics, she offers no real bibliography for the pursuit of these 'chapter headings'.

I feel that as yet I have only just scratched the surface. Thank you in advance for any considerations you can give these requests.

Bob Nathanson  
Chairman (and only member)  
Computer Science Dept.  
Kings Park Senior High School  
Kings Park, NY 11754



## 1421 Year Old Dragon

As project supervisor for L.O.C.U.S.T. (Loosely Organized Computer Users, Sunset Triad) I was pleased to hear that someone else had come up with the idea of constructing a spacecraft simulator for spacewar and similar games. My device would use modified TV typewriter circuitry and the local 2000F timesharing system. I would be interested in talking to anyone who has ideas along similar lines.

I would also be interested in any low-cost terminal rental or sales (rental preferably below \$100 a month, sales below \$200) that anyone might know of. Also, we are beginning to look into a remote controlled robot project.

On the subject of dragons, I am presently 1421 years old the recipient of hatch certificate no. 19023. I am however, unfortunately unable to prove this as the records were destroyed in a fire in 400 BC in Athens, Greece.

Bill Leininger  
219 N. School St.  
Mt. Prospect, ILL 60056

*But MITS + 400  
is 2375???*



The BIT USERS ASSOCIATION is an organization of computer hobbyists and small systems engineers serving primarily the Minneapolis, St. Paul metropolitan area. The members seek to exchange newsletters and technical data with organizations or individuals with similar interests.

The BIT USERS ASSOCIATION was begun in 1973 by users of the BIT 480-483 minicomputers, and remains source for replacement parts and licenses regarding the said machines. It has expanded to encompass persons with interest in various minicomputers and microprocessors, and is actively engaged in the design and construction of selected peripheral devices suitable for hobbyist use and which can be built at low cost at home. Raster-scan television graphic and uniquely dense alpha-numeric television terminals, for example, are in final stages of development. We will notify your publication as hardware becomes completed and its data is available.

The sponsor of the BIT USERS ASSOCIATION is the Resource Access Center, a non-profit educational center serving community service agencies. Inquirers are requested to send self-addressed stamped envelope, but no other monies please. In Minneapolis, telephone 824-6406 (or message service, 823-8247).

Address: BIT USERS ASSOCIATION  
Resource Access Center  
3010 4th Avenue South  
Minneapolis, MN 55408

Contact Person: Richard B. Koplou

## Andy Finkel Returns

I hope you are all in good health, with glossy hide, oiled tongues, and cold noses. I haven't written in a while mainly because I am now at college, and have been kinda busy. But looking at some PCC back issues has given me the strength to write this letter.

Since the move to Ithaca, I have not been able to use BASIC. (If anyone knows of an HP2000 computer convenient to Ithaca, LET ME KNOW!!) Here at Cornell our main computer is an IBM 370/168, with PL/I, APL, FORTRAN, et al, including (ick) ITF BASIC, which is so bad, it has only been loaded ONCE in the past four years (and that just because I wanted to see it). Pretty bad set-up for a BASIC dragon to be in, hmmm.

Now, I have a program you might be interested in working out something with me on. Since the beginning (\*\* Background first \*\*) of the HP2000 series, people have been writing compilers/translators, etc. in BASIC to extend the capacity of their BASIC only systems. However, the program text editor portion of the program has always been a limited, dumb thing, with barely sufficient power to enter the re-enter individual lines. The editor part to PILOT73 is a prime example. The program I have written to correct this is an editor loosely based on the Program Text Editor developed by Control Data for their CDC6400 computer systems. This program is better, in use, than the 2000F BASIC editor. Presently I have (actually, had since I do not have access to a 2000C/F right now) the program connected to COBOL, PILOT, CATNAP, CATNIP, PDP8/S Assembly language, FOCAL, and a sort of a half finished FORTRAN II (all of these written in BASIC, by me and a few friends).

I bet you thought I forgot about paragraphs. (Well, I did, but anyway...) To give you an idea of what the program can do, (and if you have never used the Editor on a CDC 6400 operating under SCOPE using INTERCOM) I will list the commands -

TIME	SEND	SITUATE	COPY	DATE	CLOCK	REPORT
STATUS	LIBSET	DIRSET	LIBLIST	XEQ	TEACH	LOAD
LIBRARY	CATALOG	SYSBULL	LENGTH	ASSETS	SYSLOG	LOGSET
COMPILE						

(EDITING COMMANDS)

CREATE	LIST	BYE	SHIFT	RESEQUENCE	ADD
DELETE	RUN	SAVE	EDIT	SCRATCH	PURGE
FILES	FORMAT	XOFF	// = //	TAPE	LINE # = / TEXT /
/ TEXT1 / = / TEXT2 /					

If you are interested, let me know.

Also on programs... I have developed an expression analysis program which can handle more and better expressions than the BASIC 'LET' statement, including more functions, N!, a root function, arrays, up to four character variable names, and all kinds of other interesting things. This program is meant to be the analysis section of compiler type programs written in BASIC, and works pretty well.

In the games dept. I have a version of DATABUS' Computer/Panzer up and running in BASIC. If you haven't done it, and want it, let me know and I'll send it along. And if you want a version of David Ahl's LOVE program translated to HP, or a lettered tape punch program, or even an inter-terminal communications program, also let me know.

Andy Finkel  
623 Mary Donlon Hall  
Cornell University  
Ithaca, NY 14853

*Andy -  
send everything*

## From Mere Egg to...

The first issue of PCC I received was enclosed with my Altair Computer Notes. I was a-maze-d. I subscribed. About four months ago when I decided to learn something about computers and sent for the MITS kit, I was truly naive and (in Dragon terms) a mere egg. Today I have the Altair and SWTP TV Typewriter and have grown in ways which a few months ago were unknown. I attribute this growth to you kind people. Thanks.

Ron Santore  
1957 Huasna Dr.  
San Luis Obispo, CA 93401

## A Grab Bag of Ideas

It's time for my bi-issue letter again, so I thought I'd dash off a note to you regarding your last issue. I have a couple of ideas for INCHWORM. Have you ever played the game "Tripples" Originally it was designed for two players, starting in opposite corners. The idea is to move onto your opponents starting base, alternating turns. The twist is that you can only move to squares that the arrows (all of the squares have randomly placed arrows on them, pointing to squares above, below, on an upward diagonal etc.) under your opponents position. It is but a simple matter to either rip off the entire game for use with two players and two Inchworms, OR make it a one player game by positioning the arrows as previously described. The latter makes it more of a challenge to go to the flowers, then around the tree, etc. My second idea is also not completely original, but it should suffice - remember that teaser game in the last issue? Well, just place Inchworm on the teaser grid. This would greatly increase the difficulty of arriving at the winning combo. Also, you could just make one big teaser board, with the "ones" representing mountains. The board would be continually changing.

I REALLY hope that the new east coast PCC will be in Philadelphia (p. 20). If so (or even [sigh] if not so) could you give me the addresses of the people to contact. Things are looking up however, computer-wise. There are some Boy Scout-type Explorer posts around me; one has access to two PDP-11's, a 1401, plus some other bigger ones - the other one has atleast a PDP-11 (with BASIC), and also might get a free charge number from somewhere and so I probably will be able to use something. The only problem is that both of the advisors that teach us things have some hang-up about the simpler languages; at one we're learning machine language, at the other AUTOCODER. I hate both with a passion. The languages, that is.

In the Scientific American of October or November, the 'Mathematical Games' section had a section on card games or tricks or something. One thing discussed gave me an idea for computer art. It went like this: You take a deck of all of one suit (say, the 13 spades) shuffle them so they are mixed up. Then you look at the top card, and count off that number of cards (A=1, K=13, Q=12 etc.). You count them off so that they are reversed (someone there must know how to do this) and put them back on the top of the deck. Usually, the cards will get arranged in order (A up to K) after some time of doing this. Once the King gets to be the bottom card, it will stay there forever. Sometimes you get an Ace on the top, ending the game. Getting to the art angle, here's what you do. Form (or input) a vector of numbers, no two the same, integers, like the cards discussed above. It is an easy thing for the computer to reverse the correct number of numbers in the vector. This is then translated into a string of symbols and spaces which the computer prints. The process goes on for x number of times. The end result should/might/could/can be an interesting pattern. I have (see above) not had an opportunity to try it out yet. You could probably dream up an idea about what to do with the 'one' or ace when it gets to the top (like counting it as a 2; putting it in a random position: putting it at the bottom of the pile, etc.). If you want to put this in your mag, just put my name somewhere, it's free.

Sorry that I keep sending you all these 1/2 baked ideas, and things, but I can't do anything much with them since they are either way over my head (can't even figure out how LUNAR works), or I can't get a hold of a terminal (like I said, my high school is too cheap); so I figure why not let someone try to do something with them that benefits mankind.

Steve Follmer  
623 Coram Rd.  
Huntingdon Valley, PA 19006

## Definitely Qualified

I read in one of the issues of the PCC newspaper that you give free subscriptions for "Datamation" to qualified persons, so . . . I am 11 1/2 years old and I can program in BASIC, a little FORTRAN, and in COBOL and machine language. I have access to a PDP-8 system 25 (BASIC language) computer and my father and I are building an Altair 8800 computer. It will only have machine language once we get it built, but we are buying a BASIC language package for it. I have written about 100 programs just in BASIC, about 22 FORTRAN, 16 COBOL, 20 in machine language so far.

I plan to be doing a lot more programming in the near future, and I am always looking for new ideas for programs.

Am I qualified?

Kurt Inman  
350 Nelson Rd.  
Scotts Valley, Ca. 95066

## HNEFATAFL?

In the September PCC, Joe McCarty was asking about programs that spell. A typical spelling program is described in "A Technique for Computer Detection and Correction for Spelling Errors," by Fred J. Damerau, *Communications of the ACM*, March, 1974.

If Joe likes heuristic TIC TAC TOE he ought to love heuristic GO (See H. Remus, "Simulation of a Learning Machine for Playing GO," *Inform, Process*, 1962. N. Holland Publ. Co., Amsterdam, 1963 p. 428-431) And for a real beta-boggler he could try his hand at heuristic HNEFATAFL. A fascinating game of incalculable antiquity and inestimable ubiquity, HNEFATAFL was a popular pastime during the reign of Athelstan (see Robinson J. Armitage's definitive *Time of St. Dunstan* for further info).

MOSFET VAN MICROBUS  
THE MACADAMIA OF ACADEMIA



### EVOLVING TECHNOLOGY SEMINARS

presents a three day short course  
"MICROPROCESSORS -  
UNDERSTANDING  
AND APPLYING THE INTEL 8080"  
Oct. 20, 21 & 22, 1975  
San Diego, California  
For information call:  
Dr. Donald Rauch (714) 224-3785

### Introducing "THE COMPUTER STORE" FEATURING ALTAIR 8800 COMPUTERS

- KITS IN STOCK
- ASSEMBLY ADVICE
- SOFTWARE INFORMATION
- BOOKS ON COMPUTERS

ARROWHEAD COMPUTER CO.  
11656 Pico Boulevard, W. L.A.  
(213) 478-3168

## News from ???

Enclosed are a couple of ads from the Los Angeles IEEE *Bulletin*. Maybe your L.A. subscribers would like to know about THE COMPUTER STORE and the 8080 seminars. Or maybe these people would like to advertise in PCC. Also enclosed is a clipping on video games, from the *Wall Street Journal*.

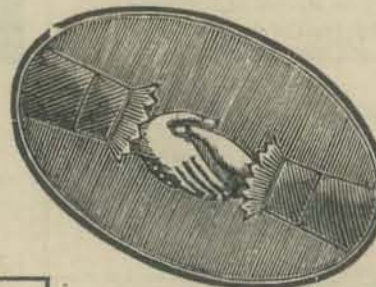
If Tony Audas (PCC, Vol. 4, No. 2, pg. 29) wants to do the I Ching routine, he might want to contact professor Dan Rie of the Wharton School of Business University of Pennsylvania. Rie has an I Ching program running, complete with Fu Shi Circle (whatever that is).

As you probably know, the literature on John Horton Conway's game of LIFE is quite extensive, and the game has been programmed not only in several dialects of BASIC but also APL, FORTRAN, PL/I, COBOL, and BAL. There is even a "structured" LIFE program in the book *Top-Down Structured Programming Techniques* by C. L. McGowan and J.R. Kelly. Any day now I expect to hear of a version written in RPG, XPL, or SNOBOL! LIFE fans looking for more information can consult Robert Wainright, 1280 Edcris Road, Yorktown Heights, NY 10598. He publishes (or used to) LIFELINE, a quarterly newsletter devoted to current developments in LIFE.

Many educational and recreational computer users seem unaware that there are quite a few places from which they can get all kinds of inexpensive programs, often paying only the cost of the source tape and/or the listings. Such sources include many universities, the HP Clearinghouse, and the HP Contributed Library, NASA, NBS, and various user groups. Most of the programs are in standard FORTRAN, COBOL, or BASIC and can usually be run on most computers with a minimum of tinkering (although not always). The Government Printing Office publishes *Computer Program Abstracts* (a quarterly catalog of NASA software) and other software items are sometimes listed in the *Monthly Catalog of US Govt. Pubs*. Professional educators and research workers may find useful items listed in the *Directory of Data Bases in the Social and Behavioral Sciences* (published by Science Associates/International). Lots of universities and other sources of miscellaneous programs are listed in the *International Directory of Computer and Information System Services* (published by Europa Publications, Ltd.) but no individual programs are mentioned. Unfortunately there doesn't seem to be any comprehensive directory of free software!

I wonder if the Star Trek game has been implemented in YERKISH? (See p. 94 of the October *Reader's Digest*.)

# 27



DATAMATION is published monthly on or about the first day of every month by Technical Publishing Company, 1301 South Grove Ave., Barrington, Illinois 60010; Arthur L. Rice, Jr., Chairman of the Board; James B. Tafel, President; Gardner F. Landon, Executive Vice President. Executive, Circulation and Advertising offices, 35 Mason Street, Greenwich, CT 06830. (203) 661-5400. Editorial offices, 1801 S. La Cienega Blvd., Los Angeles, CA 90035. Published at Chicago, Ill. DATAMATION is circulated without charge by name and title to certain qualified individuals who are employed by companies involved with automatic information handling equipment. Available to others by subscription at the rate of \$24; \$40 Air Mail annually in the U.S. and Canada. Reduced rate for qualified students, \$14.

Dear Kurt:

Hopefully your name should now be on our circulation list - you're qualified. Unfortunately our circulation is handled on a computer (a fairly reliable Honeywell system) and it may take a little time before you receive your first issue. If by December you haven't received your copy, let me know and we'll check.

Your letter is appearing in the November issue. We'll send you a copy. Thanks for writing and keep up the good work.

John L. Kirkley  
Editor - Datamation

## Computer Fair

Future consumers of computer power will have an opportunity to design and describe computer projects for school, home, and society and enter them in the 1976 National Student Computer Fair. The Fair is being held at the National Computer Conference at the New York Coliseum, June 7 - 10, 1976.

Tomorrow's computer generation - today's elementary, middle and high school students - are eligible to enter projects in the Fair. According to Dr. Sema Marks, Director of Academic Computing at the City University of New York and Chairperson of the Fair, "We want students to concentrate on the social situations they know best. We'd like students to create new computer tools for home or school use, produce a work of computer art, or design and implement a computer game. We're broadening the entry categories this year to include computer art and music, short stories, science fiction, and poetry," said Dr. Marks, "to attract a new set of students to computing. Elementary students, for example, may submit drawings of a computer."

Students participating in the 1976 National Student Fair will have the opportunity to influence the more than 50,000 computer professionals expected to attend the NCC. Winning entries will be published in Creative Computing, a journal for educators and student computer enthusiasts, and in the Proceedings of the 1976 National Student Computer Fair. Prizes include an MITS Altair 8800 Computer Kit, eight \$100 United States Savings Bonds, sixteen \$25 United States Savings Bonds, and subscriptions to Creative Computing. For entry rules, application forms and any additional information contact:

1976 National Student Computer Fair  
City University of New York  
33 West 42nd Street  
New York, NY 10036



## LO\*OP Center

The instructional resource departments of our schools and colleges are bulging with extremely flexible and sophisticated equipment. Yet many classroom teachers decline to use this new technology. Why? What is keeping the video-tape cameras and computers on their shelves, accessible to only a few?

I believe that most teachers are overwhelmed by the dilemma of what to present to their students. They are aware that their material will be used by more students more often and they conclude that it must be prepared with a proportionate increase in care. In addition, the teacher is a learner on each new machine and often hesitates to step down to the role of classmate.

The solution is disarmingly simple. It doesn't matter what is presented. The first lesson is how to turn on the screen, not what is on it. We are teaching how to press buttons, insert cartridges, and aim cameras. The ability to manipulate the machinery itself is the skill our students need for their adult lives both on the job and at home. He or she who can read the manual, use the program library, flip the switches and troubleshoot when he does not get the expected response will take his place in the society of the future with ease.

Our teachers will have to struggle along beside the children for a while. There is no way they can arm their students with the information to solve the problems which they will meet in adulthood. Nothing we can display, project or record through any medium will have other than artistic or historical impact. Next year's information has not yet been discovered, nor it's tools invented, at this point in time, "the medium is the message."

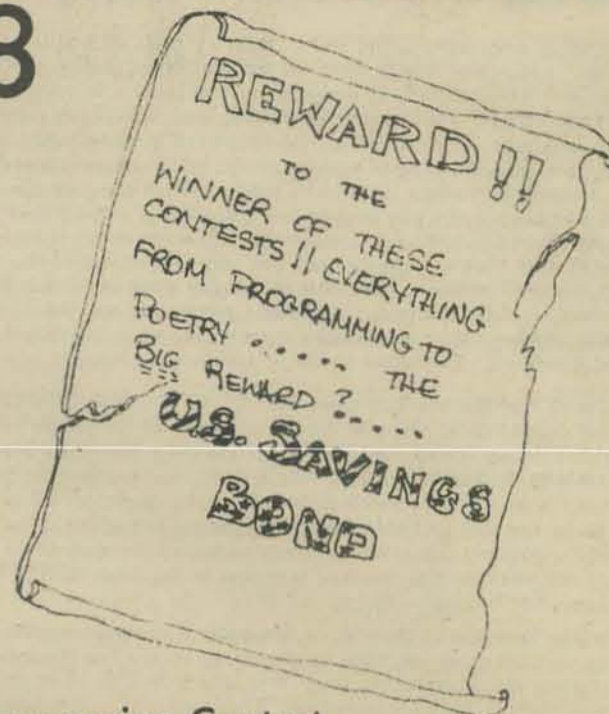
But what about the human values? If we allow technological competence to become a major goal in education will the next generation not let their appreciation for the humanities slip away? Will the human teacher be replaced by the unfeeling computer?

I think such fears are groundless. When we master the machine so that it becomes a tool we handle with a minimum of effort, it fades from our attention. Watch any child as he learns to write - at first the pencil and paper are his main focus. But soon they become extensions of his own body and mind. He concentration shifts to the drawing and the story.

Let us allow our children to openly apply themselves to the mastery of the technology they will have to face in this adult world. Let us encourage our teachers to join with them in this endeavor. Soon technology will take its proper place in the background of their lives. Human learners and teachers will then be free from the problem of "how to do it." They will turn their unique abilities to the question no machine can ever answer: "Do we want it done!"

Liza Loop  
8099 La Plaza  
Cotati, Ca. 94928

# 28



## AEDS Programming Contest

The Association for Educational Data Systems has announced plans for its thirteenth Computer Programming Contest conducted annually for students in grades 7 - 12. The Association, also known as AEDS, is a national organization comprised of administrators, teachers, systems analysts, and programmers of educational data processing systems in vocational, public, and private schools. The contest is on the approved list of national contests and activities of the National Association of Secondary School Principals.

Deadline for entries in the contest is March 1, 1976. Students wishing more details and an application blank should write to AEDS Programming Contest, Dr. Jane Donnelly Gawronski, Department of Education, San Diego County, 6401 Linda Vista Road, San Diego, CA 92111.

The Programming Contest winner will receive a \$100 US Savings Bond plus an all-expense paid trip to the 1976 AEDS Convention in Phoenix, Arizona on May 3 - May 7. The winning students sponsor also will receive an all-expense paid trip to the convention. Second prize awards of \$25 US Savings Bonds will be given. Students who submit projects also are eligible to receive a one-year subscription to a professional publication.

A project may be submitted by an individual or by a team of two or more pupils. In the event that the project winning the grand prize is submitted by a team, the team must select one of its members to make the trip to Phoenix.

The winner of the twelfth annual contest was Heinjo Lahring of Calgary, Alberta, Canada.

## Solar Collector

For anyone seriously considering a solar heating installation, the latest development of the Corning Glass Co., The Corning Tubular Evacuated Solar Collector, is well worth looking into.

This collector will permit much higher temperature operation, with a consequent reduction in required storage volume. At the high end of its temperature range it is well within the temperature required for efficient operation of absorption refrigeration equipment. For the dedicated solarite, these collectors approach the temperature rise needed to operate a vapor cycle turbo-generator. For more info, check with Corning Glass Works, Corning, NY 14833.

-O.S.- (The Old Soldier)



## Science for the People

A CPE course presented by the Tallahassee chapter of SCIENTISTS AND ENGINEERS FOR SOCIAL AND POLITICAL ACTION (SESPA).

SESPA is presently a loosely structured organization of national scope. It is made up of secretaries and scientists, technicians and teachers - all those who do and are affected by science and technology. To get an idea of what SESPA is about check out their periodical SCIENCE FOR THE PEOPLE (available at CO-OP Books and Records and at SESPA meetings).

The CPE course will include presentations and discussion about socially relevant scientific topics. Among other things it will analyze the role of science and scientists in society.

For information about the meeting time, place, discussion topics, etc. call:

222-7080 ask for Bob

224-1617 ask for Jim

The course is oriented to both scientists and non-scientists. Please join us.

Tallahassee SSPA  
Progressive Technology  
P. O. Box 20049  
Tallahassee, FL 32304

## PRINCIPLES AND PRACTICE OF ELECTRONIC MUSIC

by: Gilbert Tryhall  
Grosset and Dunlop, 1973, 214 pgs., \$6.95

The first thing that strikes you as you pick up this book is that it is terribly stiff for being paper bound. Only upon further examination do you discover that the stiffness is due to the presence of a 7-inch 33-1/3 rpm record attached to the inside of the back cover. If you lose control and play the record before reading the chapter of the book that goes with each band, you may not find the disc too meaningful. It is full of a lot of petty buzzes and squeaks, all introduced by a velvet-voiced narrator as though they were going to be something spectacular.

If you restrain yourself and read the book first, the noises on the record fall right into place. Electronic music is as new to electronics as it is to music, and whether you are new to electronics, new to music, or new to both, Tryhall will take your hand and lead you on an enjoyable and eminently uncluttered journey into the land that is the upshot of today and the downbeat of tomorrow. Every time the book describes how a sound of a particular character is made and what the wave form looks like, the record lets you listen to it. This added feature brings the book to life in a way no bolt of lightning could have done for Dr. Frankenstein's monster, and indeed, takes it out of the league of mere books and makes it an educational program.

Gilbert Tryhall is a musician. He, therefore, conducts this program as though he were introducing other musicians to a new instrument, or perhaps to a new genre of music. He treats electronics gingerly and with great reverence, as though trying to allay the fears of the hitherto non-electronic musician. But he knows his stuff, and doesn't tell any of the little fibs people trying to make 'trons go down easier are inclined to tell, nor does he assume so much of a musical background in his students that he befuddles the non-musician.

The book begins with an introduction to electronic music as a new musical discipline, listing some of the areas where synthesized music has gained acceptance, then previewing some of the aspects of music and of electronics which make up the world of electronic music. The first few chapters are concerned with the relationship between waveform and sound quality, and with the elementary electronic concepts one needs to know in order to produce the desired waveforms. At this point, the author takes a radical break from the theoretical and begins to build, piece by piece, an electronic music studio, a "synthesizer-on-the-hoof", if you will. With a free corner and some money, anyone who can solder (if you can't, Tryhall teaches you!) can have his own synthesis and recording facility going as fast as you can say "bankrupt".

Once you have put your electronic music studio through its paces to ensure that it is up to classical standards, you then learn how to translate conventional musical notation of all forms into synthesis instructions. If, in the course of exercising your new machinery, you stumble across a combination of sounds that amounts to a hit, Uncle Gilbert tells you how to get and protect a copyright. If it has not already become quite apparent, Tryhall leaves nothing to chance.

Finally, there is a brief history of electronic music, beginning with stuff that was really "electricity music" and electro-mechanical, and proceeding through present analog and digital techniques, to come to rest on a bright outlook for the future. True to form, there is a 200-word glossary of electronic music terminology at the end.

The very least one could say about this book is that it is complete. It gives equal time to both technical and creative aspects of music synthesis, and lacks neither detail nor clarity in either area. Best of all, this enormous glut of data is presented with such order and enthusiasm that comprehending even the most esoteric aspects (for Tryhall, the electronics; for me, the music) becomes a joy in itself.

Howard A. Hurtt  
Sacramento, Ca.

## A NEW MUSICAL DISCIPLINE

AVAILABLE FROM: PCC BOOKSTORE Principles and Practice of Electronic Music  
and  
Bug Books I & II (set only)



## LOGIC & MEMORY EXPERIMENTS Using TTL Integrated Circuits

By: Peter R. Rony, Dept. of Chemical Engineering  
David G. Larsen, Dept. of Chemistry  
with assistance by  
Roberts A. Braden, Dept. of Learning Resources  
Virginia Polytechnic Institute & State University

E & L Instruments, Inc. 1974, two volumes, \$16.95

Interesting to note that books are written by two professionals with Chemistry and Chemical Engineering backgrounds (probably adds to their ability to approach the subject more openly and perhaps more in tune with the learner).

Attention that has been paid to ease of assembly and disassembly as one moves from experiment to experiment. The detailed efforts by the designers show not only in the electronic nature of the product but also in the mechanical aspects of the kits (e.g. velcro pads to hold modules in place).

Logical and detailed presentation

Figure 1-15. A complete breadboarding system doesn't require much hardware. A socket, battery, some chips, wire, resistors, capacitors, and tools will suffice.



VOLTAGE SOURCE

+ 5 V

POWER SUPPLY

12 V

BATTERY

+

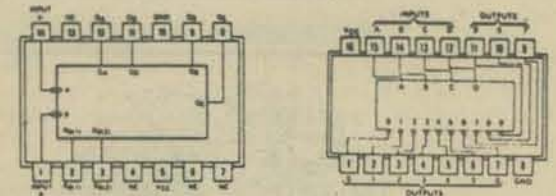
12 V

RESISTOR

330 kΩ

SYMBOLS FOR AUXILIARY FUNCTIONS AND COMPONENTS

PIN CONFIGURATIONS OF INTEGRATED CIRCUIT CHIPS



Much of the "dog-work" of putting the less glamorous sub-modular together has already been done. This adds to the ability of the experimenter to concentrate more fully on the interesting material to be learned... and to spend his time on significant experiments.

comments on Bugs Books by Dan Pichulo

### Into Electronic Music?

If you have always dreamed of building and playing your own synthesizer, improving upon it and watching it grow more impressive day by day as it consumes your living space and bank account but enriches your culture, consider contacting:



B. A. Hutchins  
Musical Engineering Group  
203 Snyder Hill Road  
Ithaca NY 14850

The Musical Engineering Group, formerly the Electronic Music Club, puts out an innovative, technical, hardware-oriented newsletter called *Electronotes*, which has been in publication since January of 1972. The no-frills bi-weekly is a treasure trove of state-of-the-art theory, projects and hardware access information. If you subscribe and start ordering back issues, you will be able to catch up to the latest in electronic music engineering (a speed-reading course would be helpful, as the volume of information is prodigious) and also to trace the fascinating evolution of the *Electronotes* logo. Write for the latest subscription rates.

Howard A. Hurtt  
Sacramento, California

## Active-Filter Cookbook

By Don Lancaster  
Howard W. Sams & Co., Inc., 1975, \$14.95

An active filter is a neat goodie, right? An active filter is easy to make if you know how, right? Right! How do you find out how to design an active filter? Until now you looked in technical publications, manufacturers' application notes, and obscure chapters in text books.

Don Lancaster, the author of *RTL Cookbook* and *TTL Cookbook*, has produced a single volume that shows how to select the filter type that you want and then how to design it. In keeping with the past efforts the title is *Active Filter Cookbook*.

This book covers the basics of filters and operational amplifiers, filter types and response curves, and how to design filters. There is a chapter on tunable filters.

Using this book, it is a simple task to select a circuit that fits your needs and, using simple math, select component values to give the desired characteristics.

The information is presented at an elementary enough level that, if you know that you are interested in active filters, you can use the book. Illustrations are used to give a quick summary of each filter circuit.

I have three complaints about this book. One- The explanation of the complex variables is that they are complex variables. Two- The price is high \$14.95. Three- It wasn't published sooner.

Robert Mullen  
Castro Valley, Ca.



Buy a Mouse Trap

*Robert Mullen*

## ELECTRONIC PROJECTS for MUSICIANS

By Craig Anderson  
Guitar Player Publications, 1975, \$6.95

Electronic Projects for Musicians is a new book, written by my friend Craig Anderson, which I heartily endorse for all electronic and musical types. The first 4 chapters tell how to obtain and identify parts, select and care for tools, and apply basic construction techniques; in short, an introduction to basic electronics a la Radio Amateur's Handbook. Chapter 5 contains 19 projects for musical/audio applications, and the book concludes with sections on troubleshooting and where to find more information.

I believe the book to be of equal interest to the neophyte and to more experienced electronic types. For the neophyte with musical know how the book provides a very understandable, practical, and readable insight into the world of electronics-- which can do so much to extend his or her capability. For those already versed in electronics, this book opens the door to the world of musical and electronic effects. I'm very enthused about this book and have enjoyed reading portions of it during it's creation; I believe it's an ideal primer for anyone interested in music/electronics regardless of their age or experience.

Bill Godbout  
Oakland, Ca.

Godbout Electronics, Box 2355, Oakland Airport, Ca. 94614, supplies parts kits for all of the 19 projects in *Electronic Projects for Musicians*.



HELP! (cried the Dragon)

## New PCC Publications

I'm overwhelmed... too much information to stuff into PCC. Each issue **could be** all letters, or all games in BASIC, or all machine language stuff about the ALTAIR, or stories about people using computers in schools, or listings of interesting programs or circuit diagrams, or how to buy a personal computer, or energy stuff, or...

And... people are sending us reams of paper about TINY BASIC.

And... for 13 years, I have been accumulating ideas about kids learning how to use computers. In one corner of our office sits a big, clunky Xerox machine. Chug, chug, chug, it grinds out copies. Everyday, I look at it quizzically, wondering how to use it in some interesting and creative way.

So... we will crank up our Xerox and start (1) a newsletter called TINY BASIC and (2) a series of booklets called COMPUTERS IN THE CLASSROOM.

TINY BASIC is about building your own BASIC and... maybe... it might grow into a publication about learning how to build your own software for the home/school/personal computer. We will publish ideas and software in the public domain for anyone to use. TINY BASIC will be a sharing thing with explicit permission to use the information in a non-commercial way. We will start modestly, promising 3 issues over the next few months. Bernard Greening is the editor of TINY BASIC with help from Dennis Allison and whoever wants to help.

TINY BASIC, volume 1. Three issues for \$3.  
First issue: January 1976.

Send your \$3.00 to TINY BASIC, P.O. Box 310, Menlo Park, Ca. 94025.

[ And! Send us some TINY BASIC stuff to publish.]



COMPUTERS IN THE CLASSROOM will be a series of Xeroxed booklets, mostly about BASIC. Learning BASIC, teaching BASIC, using BASIC at school, at home, wherever. The first booklet is about BASIC in secondary school math. Ideas, problems, games, listings of programs, sources of information, workbook pages you can copy on your Xerox and hand out to your students.

COMPUTERS IN THE CLASSROOM, Book 1. Available Jan. 1976.  
Xeroxed, 60 pages. \$3.00  
from PCC Bookstore

P.S. COMPUTERS IN THE CLASSROOM is the name of a University of California extension course that LeRoy Finkel & Jerry Brown & I have "taught" during the last 6 or 7 years at UC Berkeley, UC Davis, UC Riverside, UC Santa Cruz, UC San Diego, UC Irvine, UC Santa Barbara. About 1000 elementary and secondary school teachers have meandered through these courses.



Round Dragon

Both of these Chinese papercut dragon designs are available in 24" cotton squares, hand silkscreened in beautiful colors and ready to hang on any wall.

\$8.00 each

30



COMING SOON

American Indian Society of Washington, D.C.  
Cookbook

Send \$4.95 plus 50¢ postage and handling to:

American Indian Society  
519 - 5th St. S. E.  
Washington, D. C. 20003



Prairie Fire Dragon

They are also available screened on glass and mirrors, 12" size, mounted and ready to hang.

\$10.00 each

From: Alan Grinberg  
1360 Howard St.  
San Francisco, Ca. 94103  
(415) 863-1207

# New Titles

*see Book Reviews  
pages 29 & 30*

\* **BUGBOOK I & II (SET ONLY) \$16.95**

PRINCIPLES & PRACTICES OF  
\* **ELECTRONIC MUSIC \$6.95**

\* **ACTIVE-FILTER cookbook 14.95**

\* **COMPUTERS in the CLASSROOM \$3.00**

\* **ELECTRONIC PROJECTS for  
MUSICIANS \$6.95**



ANSWERS TO PUZZLE ON PAGE 21

I	N	T	E	R	F	A	C	E	B	U	S	S
N	U	M	B	E	R	A	T	F	K	O	W	
T	L	B	S	A	R	C	A	N	S	I		
E	L	A	S	T	I	C	S	B			T	
L	S	I	S	L	Y	K	A	R	N	A	C	
						B	E	S	E	E	C	H
C	A	B	O	B	E	I	I	L	E	N	E	
A	N	G	R	Y	N	T	C	Y	T	E	S	
S	T	U	N	T	E	D						
S	I	N	G	E	R	R	A	M	M	S	L	
E			S		H	I	R	O	H	I	T	O
T	I	T	L	S	T	O	D	R	A	O		
T	E	E	D	C	B	E	A	R	L	A	P	
E	N	D	S	H	O	B	B	Y	I	S	T	S

RON SANTORE

## PCC Bookstore

- BASIC, Albrecht, Finkel & Brown, 1973, p.323 - \$3.95
- Basic BASIC, James Coan, 1970, p. 256 - \$7.45
- BASIC PROGRAMMING, Kemeny & Kurtz, 1967, p. 145 - \$8.50
- COMPUTERS & COMPUTATION, Scientific American, p. 280 - \$6.00
- COMPUTER LIB & DREAM MACHINES, Theodore H. Nelson, 1974, p. 186 - \$7.00
- DRAGON SHIRTS, Nancy Hertert, 1974 - \$3.50
- GAMES, TRICKS & PUZZLES, Wallace Judd, 1974, p. 100 - \$2.95
- GAMES WITH THE POCKET CALCULATOR, Thiagarajan & Stolovitch, 1975 - \$2.00
- MATH, WRITING & GAMES, Herbert Kohl, 1974, p. 252 - \$2.45
- MY COMPUTER LIKES ME, Bob Albrecht, 1972, p. 64 - \$2.00
- 101 BASIC GAMES, Ed. David Ahl, 1974, p. 250 - \$7.50
- PROBLEMS FOR COMPUTER SOLUTION, Gruenberger & Jaffray, 1965 - \$7.95
- PROFESSOR GOOGOL, Sam Valenza, Jr., 1973, p. 144 - \$3.25
- PROBABILITY, D.J.Koosis, 1973, p. 163 - \$2.95
- PCC GAMES Program Listings - \$2.00
- PRACTICAL, LOW-COST HOME/SCHOOL MICROPROCESSOR SYSTEM, 1974 - \$1.00
- STEPS TO AN ECOLOGY OF MIND, Gregory Bateson, 1972, p. 517 - \$1.95
- STATISTICS, D.J.Koosis, 1972, p. 282 - \$3.95
- THE ENERGY PRIMER, Portola Institute, 1974, p. 200 - \$5.50
- TTL COOKBOOK, Don Lancaster, 1974, p. 328 - \$7.95
- II CYBERNETIC FRONTIERS, Stewart Brand, 1974, p. 96 - \$2.00
- WHOLE EARTH EPILOG, Stewart Brand, Editor, 1974, p. 318 - \$4.00
- WHAT TO DO AFTER YOU HIT RETURN, PCC, 1975, p. 157 - \$6.95
- DIGITAL LOGIC CIRCUITS, Sol Libes, 1975, p. 184 - \$5.98

**31**

### TO ORDER—

name \_\_\_\_\_  
address \_\_\_\_\_ zip \_\_\_\_\_

Quantity	Item	Price Each	Total

\* Please add \$0.50 for orders under \$10.00 and \$1.00 for orders \$10.00 and over.

Sales Tax	
Postage*	
Total	

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

# People's Computer Company

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

## Subscriptions

Name \_\_\_\_\_  
Address \_\_\_\_\_

Zip \_\_\_\_\_

Regular - 1yr \$5  2yr \$9

Group (how many) \_\_\_\_\_

### Back Issues

Vol 1 \$1

Vol 2 \$1

Vol 3 \$4

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

Subscriptions are \$5 for 6 issues. (\$6.00 outside the U.S.A. — surface mail; \$12.00 air mail.) Subscriptions begin with the July issue.

NEW STUFF! See Page 10 for announcements of TINY BASIC Newsletter and COMPUTERS IN THE CLASSROOM booklets about kids, teachers and BASIC.

## Gift Subscription

Gift Recipient's Name and Address
_____
_____
_____
Your Name and Address
_____
_____
_____
Send Gift Card and Sign
_____

# Have A Beautiful Holiday

Editor BOB ALBRECHT

Production MARY JO MC PHEE

Circulation LOIS BRITTON

Contributors DENNIS ALLISON  
ROBIN ALLISON  
KEITH BRITTON  
OLAS DE PAUL  
BERNARD GREENING  
DEAN KAHN  
ROBERT MULLEN  
MAC MULLEN

Contents

1	Subscription Information
2	Kids Building Kits
4	NUMBER — In 3 Kinds of BASIC
6	NUMBER — In 8080 Assembly Language
8	U.S. Schools in Europe
9	Talking Calculator
10	Tiny BASIC
13	Christmas Buyer's Guide / Tarbell
14	Cromemco / Swtp
15	Godbout / MITS
16	Sphere / Xebec / Phideck / Jolt / Mike
17	Processor Technology / Logical Machine
18	IMSAI / Snat / Blastmasters
19	Analog to Digital Conversion
20	Clubs and Pubs.
21	PCC's First Crossword Puzzle
22	Biofeedback and the Arts
23	Minuet in G Minor / How to Build a House
24	FORTTRAN MAN
25	Letters and Stuff
29	Book Reviews
31	PCC Bookstore

