Build a Z8-Based Control Computer with BASIC, Part 2

Steve Ciarcia
POB 582
Glastonbury CT 06033

The Z8-BASIC Microcomputer system described in this two-part article is unlike any computer presently available for dedicated control applications. Based on a single-chip Zilog Z8 microcomputer with an on-board tiny-BASIC interpreter, this unit offers an extraordinary amount of power in a very small package. It is no longer necessary to use expensive program-development systems. Computer control can now be applied to many areas where it was not previously cost-effective.

The Z8-BASIC Microcomputer is intended for use as an intelligent controller, easy to program and inexpensive enough to dedicate to specific control tasks. It can also serve as a low-cost tiny-BASIC computer for general interest. Technical specifications for the unit are shown in the "At a Glance" box on page 52.

Last month I described the design of the Z8-BASIC Microcomputer hardware and the architectures of the Z8671 microcomputer component and Z6132 32 K-bit Quasi-Static Memory. This month I'd like to continue the description of the tiny-BASIC interpreter, discuss how the BASIC program is stored in memory, and demonstrate a few simple applications.

Process-Control BASIC

The BASIC interpreter contained in ROM (read-only memory) within the Z8671 is officially called the Zilog BASIC/Debug monitor. It is essentially a 2 K-byte integer BASIC which has been optimized for speed and flexibility in process-control applications.

There are 15 keywords: GOTO, GO@, USR, GOSUB, IF...THEN, INPUT, IN, LET, LIST, NEW, REM, RUN, RETURN, STOP, PRINT (and PRINT HEX). Twenty-six numeric variables (A through Z) are supported; and numbers can be ex-

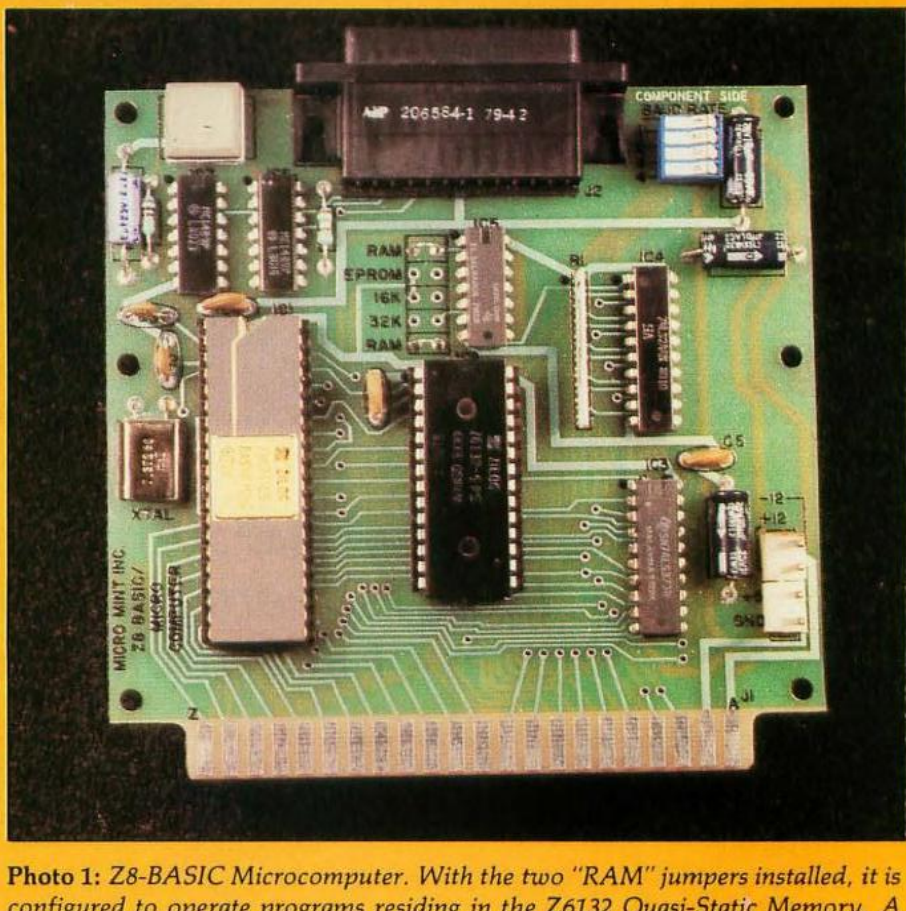


Photo 1: *Z8-BASIC Microcomputer. With the two "RAM" jumpers installed, it is configured to operate programs residing in the Z6132 Quasi-Static Memory. A four-position DIP (dual-inline pin) switch (at upper right) sets the serial data rate for communication with a user terminal connected to the DB-25S RS-232C connector on the top center. The reset button is on the top left.*

NEVER UNDER$OLD.

Thats right. If you can find a lower price in this magazine WE WILL BEAT IT! Period.*

S-100 HARDWARE

SD SYSTEMS	
Versa Floppy I kit	$259
Versa Floppy II kit	$349
ExpandoRam II kit	$249
ITHICA INTERSYSTEMS	$CALL
ALTOS	$CALL
NORTHSTAR	$CALL

SUPPLIES

Verbatim Datalife Diskettes	
MD525-01,10,16	$26.50
FD34-9000	$30.00
FD34-8000	$44.00
Avery Tabulables	
5,000 3½ X 15/16	$18.75
3,000 3½ X 15/16	$14.25
1,000 3½ X 15/16	$8.15
UARCO Paper (prices are FOB S.P.)	
9½ X 11, 15 or 18 Lb.	$25.95
14 7/8 X 11, 15 or 18 Lb.	$35.00

SOFTWARE

Microsoft Basic-80	$299
Microsoft Basic Compiler	$319
Microsoft Fortran-80	$399
Magic Wand	$275
Apple Visicalc	$113
Pickles and Trout CP/M®	$175

APPLE HARDWARE

Microsoft Z-80 Softcard	$259
Microsoft Ramcard	$170
SSM AIO A&T	$165
SYMTEC Apple Light Pen	$219
MOUNTAIN HARDWARE:	
Super Talker	$270
ROMWRITER	$157
INTROL/X-10	$180
ROMPLUS+	$162
MUSICSYSTEM	$499
Apple Clock	$252
Lobo Drive	$CALL
Videx 80 x 24	$299
Andromeda	$170
M & R Sup-R-Term	$299
Enhancer	$100
NEC Green Monitor	$240
Sanyo Green Monitor	$240
D.C. Hayes Micromodem	$299
CALIFORNIA COMPUTER SYSTEMS	

See our full page ad in this magazine.

ALPHA BYTE STORES

PRINTERS

EPSON MX-80	CALL
MX-80 Graphics ROM	CALL
NEW EPSON MX-80 TF	CALL
Okidata Microline 80	$415
Okidata Microline 82	$625
Okidata Microline 83	$899
IDS Paper Tiger 460G	$1050
Anadex DP-9000,1	$1195
Anadex DP-9500,1	$1295
Centronics 737	$689
NEC Spinwriter	$CALL
NEC w/ Sellum Option	$CALL
C. Itoh Starwriter 25 cps	$1440
C. Itoh Starwriter 45 cps	$1770
Diablo 630	$2195
Malibu 165	$1995
Malibu 200	$CALL
MPI 88G	$495

TRS-80 HARDWARE

TRS-80's	$CALL
Micropolis 77 track	$399
Percom Doubler	$199
Percom Separator	$27
Orchestra 80	$79

5115 Douglas Fir Road • Suite B • Calabasas, CA 91302

*Include $2.00 shipping and handling on all orders. Visa and Master Card accepted. Never undersold offer applies only to items currently deliverable from other retail vendors at advertised price. All never undersold offers good as supply lasts. Please add $2.00 for all COD orders. Please call for items not listed. We gladly answer any questions on all of our hardware, software, and supply needs. Quanitity discounts available. School purchase orders accepted. Please remember to figure competitors shipping and handling charges when arriving at never undersold price.

(213) 883-8594

pressed in either decimal or hexadecimal format. BASIC/Debug can directly address the Z8's internal registers and all external memory. Byte references, which use the "@" character followed by an address, may be used to modify a single register in the processor, an I/O port, or a memory location. For example, @4096 specifies decimal memory location 4096, and @%F6 specifies the port-2 mode-control register at decimal location 246. (The percent symbol indicates that the characters following it are to be interpreted as a hexadecimal numeral.) To place the value 45 in memory location 4096, the command is simply, @4096=45 (or @%1000=%2D).

Command abbreviations are standard with most tiny-BASIC interpreters, but this interpreter allows some extremes if you want to limit program space. For example:

```
IF 1>X THEN GOTO 1000
      can be abbreviated
IF 1>X 1000

PRINT"THE VALUE IS ";S
      can be abbreviated
"THE VALUE IS ";S

IF X=Y THEN IF Y=Z
THEN PRINT "X=Z"
      can be abbreviated
IF X=Y IF Y=Z "X=Z"
```

One important difference between most versions of BASIC and Zilog's BASIC/Debug is that the latter allows variables to contain statement numbers for branching, and variable storage is not cleared before a program is run. Statements such as GOSUB X or GOTO A*E−Z are valid. It is also possible to pass values from one program to another. These variations serve to extend the capabilities of BASIC/Debug.

In my opinion, the main feature that separates this BASIC from others is the extent of documentation supplied with the Z8671. Frequently, a computer user will ask me how he can obtain the source-code listing for the BASIC interpreter he is using. Most often, I have to reply that it is not available. Software manufacturers that have invested many man-years

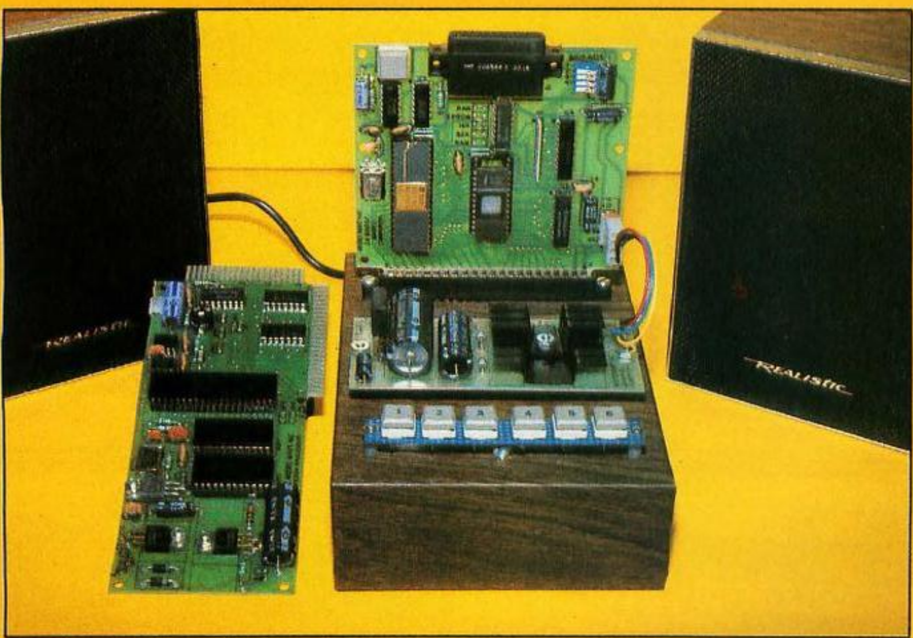


Photo 2: *The Z8/Micromouth demonstrator. A Z8-BASIC Microcomputer is configured to run a ROM-resident program that exercises the Micromouth speech synthesizer presented in the June Circuit Cellar article. A Micromouth board similar to that shown on the left is mounted inside the enclosure. Six pushbutton switches, connected to a parallel input port on the Z8 board, select various speech-demonstration sequences. The Micromouth board is driven from a second parallel port on the Z8 board.*

At a Glance

Name
Z8-BASIC Microcomputer

Processor
Zilog Z8-family Z8671 8-bit microcomputer with programmable (read/write) memory, read-only memory, and I/O in a single package. The Z8671 includes a 2 K-byte tiny-BASIC/Debug resident interpreter in ROM, 144 internal 8-bit registers, and 32 I/O lines. System uses 7.3728 MHz crystal to establish clock rate. Two internal and four external interrupts.

Memory
Uses Z6132 4 K-byte Quasi-Static Memory (pin-compatible with 2716 and 2732 EPROMs); 2 K-byte ROM in Z8671. Memory externally expandable to 62 K bytes of program memory and 62 K bytes of data memory.

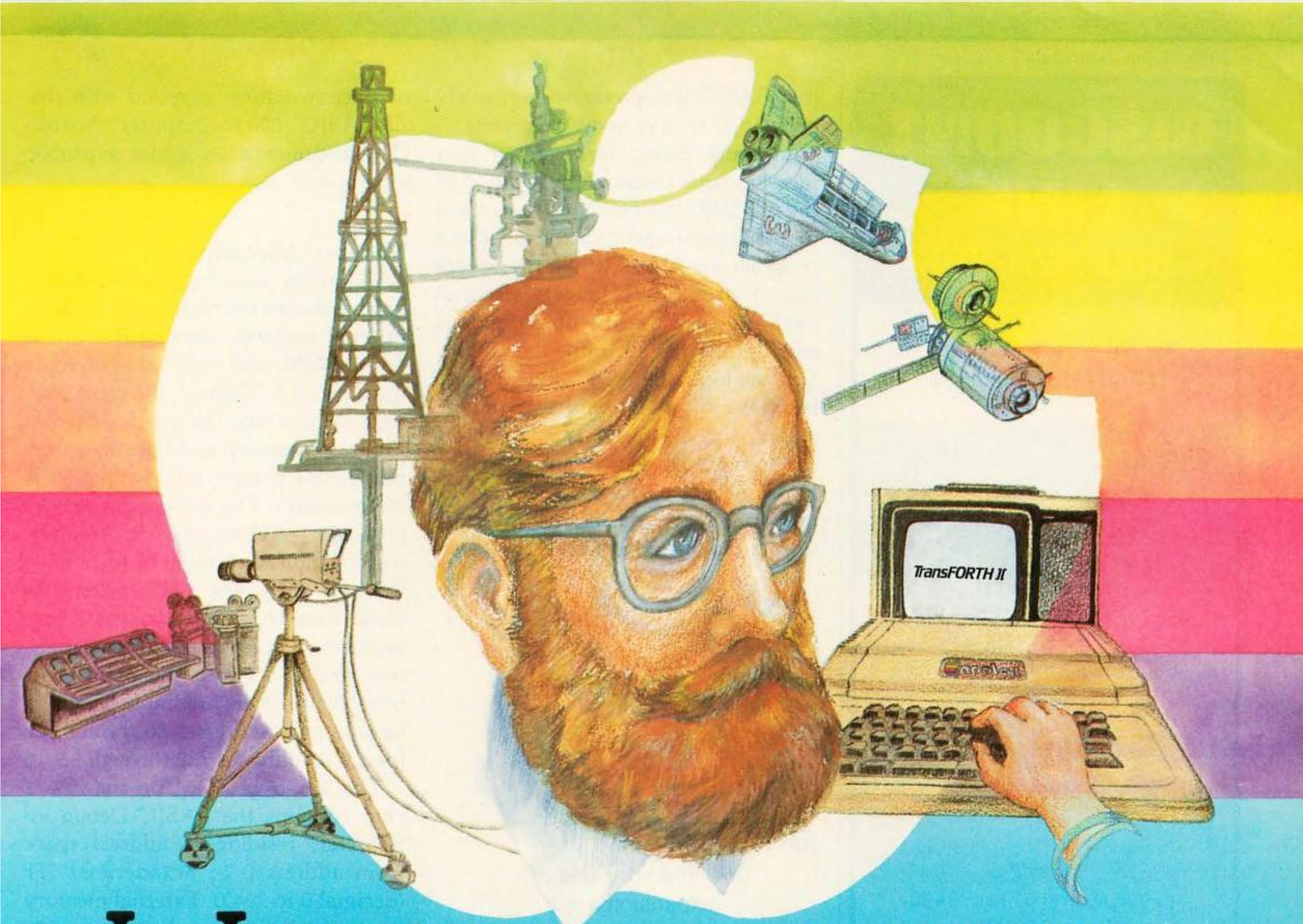
Input/Output
Serial port: RS-232C-compatible and switch-selectable to 110, 150, 300, 1200, 2400, 4800, and 9600 bps.
Parallel I/O: two parallel ports; one dedicated to input, the other bit-programmable as input or output; programmable interrupt and handshaking lines; LSTTL-compatible.
External I/O: 16-bit address and 8-bit bidirectional data bus brought out to expansion connector.

BASIC Keywords
GOTO, GO@, USR, GOSUB, IF...THEN, INPUT, LET, LIST, NEW, REM, RETURN, RUN, STOP, IN, PRINT, PRINT HEX. Integer arithmetic/logic operators: +, −, /, *, and AND; BASIC can call machine-language subroutines for increased execution speed; allows complete memory and register interrogation and modification.

Power-Supply Requirements
+5 V ±5% at 250 mA
+12 V ±10% at 30 mA
−12 V ±10% at 30 mA
(The 12 V supplies are required only for RS-232C operation.)

Dimensions and Connections
4- by 4½-inch board; dual 22-pin (0.156-inch) edge connector. 25-pin RS-232C female D-subminiature (DB-25S) connector; 4-pole DIP-switch data-rate selector.

Operating Conditions
Temperature: 0 to 50°C (32 to 122°F)
Humidity: 10 to 90% relative humidity (noncondensing)



In Language Lies The Future.

Paul Lutus creates language. Language for the future.
From him we have Apple Writer, Apple World and MusiComp. His software was used by the Viking Mars Lander team and other advanced space programs.
For the past three years he has applied himself to the development of more efficient and powerful computer languages to help him with his work. These new languages of the future are now available for your use exclusively from Insoft.

See the new world of programming from Insoft at your Apple dealer.

insoft

The languages of the future. The tools of today.

Apple is a registered trademark of Apple Computer, Inc.

259 Barnett Rd. / Unit 3 / Medford, Or 97501 / (503) 779-2465

TransFORTH II

The final "word" in Apple programming. A brilliant extension of FORTH.

- Fully compiled
- Floating point
- Transcendental functions
- Strings and arrays
- Hires, Lores and Turtlegraphics
- Music

Far more compact and approachable than Pascal, TransFORTH II is both recursive and structured and easier to use than BASIC.
Over three years in development, this high level language enables you to program in English with far greater speed and convenience than ever before possible.
TransFORTH II. A transformation in computer programming. Available today.

ALD System II

The Assembly Language Development System. Already the preferred assembler for professional software developers.

- Object files to 18K
- Source files to 37K
- Cursor based screen editor
- Upper and lower case text entry
- Nested macro instructions
- Local, global and universal labels
- Comprehensive error trapping

The enormous file capacity of ALD System II assures you of the ability to develop virtually any imaginable software system. And you can do so with ease since ALD System II edits with the simplicity of a word processor.
ALD System II. Powerful yet convenient. The first choice of professionals.

The best comes last. Both new programs are available for the Apple II or Apple III.

in a BASIC interpreter are not easily persuaded to give away its secrets.

In most cases, however, a user merely wants to know the location of the GOSUB...RETURN address stack or the format and location of stored program variables. While the source code for BASIC/Debug is also not available (because the object code is mask-programmed into the ROM, you couldn't change it anyway), the locations of all variables, pointers, stacks, etc, are fixed, and their storage formats are defined and described in detail. The 60-page BASIC/Debug user's manual contains this information and is included in the 200 pages of documentation supplied with the Z8-BASIC Microcomputer board. (The documentation is also available separately.)

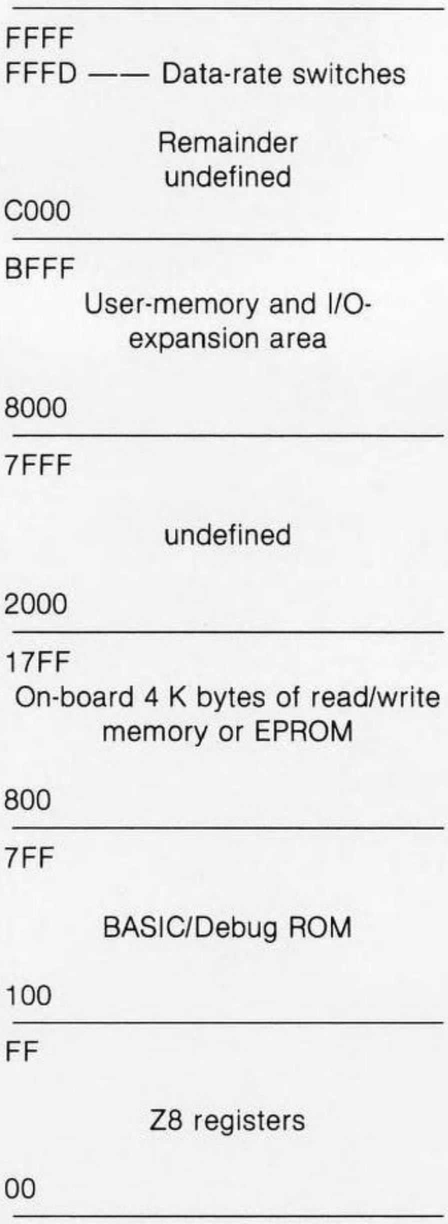


Figure 1: *A simplified hexadecimal memory map of the Z8-BASIC Microcomputer.*

Memory Allocation

Z8-family microcomputers distinguish between four kinds of memory: internal registers, internal ROM, external ROM, and external read/write memory. (A slightly different distinction can also be made between program memory and data memory, but in this project this distinction is unnecessary.) The register file resides in memory-address space in hexadecimal locations 0 through FF (decimal 0 through 255). The 144 registers include four I/O- (input/output) port registers, 124 general-purpose registers, and 16 status and control registers. (No registers are implemented in hexadecimal addresses 80 through EF [decimal addresses 128 through 239]).

The 2 K-byte ROM on the Z8671 chip contains the BASIC/Debug interpreter, residing in address space from address 0 to hexadecimal 7FF (decimal 0 to 2047). External memory starts at hexadecimal address 800 (decimal 2048). A memory map of the Z8-BASIC Microcomputer system is shown in figure 1.

When the system is first turned on, BASIC/Debug determines how much external read/write memory is available, initializes memory pointers, and checks for the existence of an autostart-up program. In a system with external read/write memory, the top page is used for the line buffer, program-variable storage, and the GOSUB...RETURN address stack. Program execution begins at hexadecimal location 800 (decimal 2048).

When BASIC/Debug finds no external read/write memory, the internal registers are used to store the variables, line buffer, and GOSUB...RETURN stack. This limits the depth of the stack and the number of variables that can be used simultaneously, but the restriction is not too severe in most control applications. In a system without external memory, automatic program execution begins at hexadecimal location 1020 (decimal 4128).

In a system that uses an external 2 K-byte EPROM (type 2716), wraparound addressing occurs, because the state of the twelfth address line on the address bus (A11) is ignored. (A 4 K-byte type-2732 EPROM device does use A11.) A 2716 EPROM device inserted in the Z6132's memory socket will read from the same memory cells in response to accesses to both logical hexadecimal addresses 800 and 1000. Similarly, hexadecimal addresses 820 and 1020 will be treated as equivalent by the 2716 EPROM. Therefore, when a 2 K-byte 2716 EPROM is being used, the auto-start address, normally operating at hexadecimal 1020, will begin execution of any program beginning at hexadecimal location 820. For the purposes of this discussion, you may assume that programs stored in EPROM use type-2716 devices and that references to hexadecimal address 820 also apply to hexadecimal address 1020.

Program Storage

The program-storage format for BASIC/Debug programs is the same in both types of memory. Each BASIC statement begins with a line number and ends with a delimiter. If you were to connect a video terminal or teletypewriter to the RS-232C serial port and type the following line:

100 PRINT "TEST"

it would be stored in memory beginning at hexadecimal location 800 as shown in listing 1.

Listing 1: *Simple illustration of BASIC program storage in the Z8-BASIC Microcomputer.*

	100		P	R	I	N	T		"	T
800	00	64	50	52	49	4E	54	20	22	54
	E	S	T	"						
80A	45	53	54	22	00	FF	FF			

Listing 2: *A multiple-line illustration of BASIC program storage.*

```
100 A=5
200 B=6
3005 "A*B=";A*B
```

	100		A	=	5		200		B	=
800	00	64	41	3D	35	00	00	C8	42	3D
	6		3005		"	A	*	B	=	"
80A	36	00	0B	BD	22	41	2A	42	3D	22
	;	A	*	B						
814	3B	41	2A	42	00	FF	FF			

The first 2 bytes of any BASIC statement contain the binary equivalent of the line number (100 decimal equals 64 hexadecimal). Next are bytes containing the ASCII (American Standard Code for Information Interchange) values of characters in the statement, followed by a delimiter byte (containing 00) which indicates the end of the line. The last statement in the program (in this case the only one) is followed by 2 bytes containing the hexadecimal value FFFF, which designates line number 65535.

The multiple-line program in listing 2 further illustrates this storage format.

One final example of this is illustrated in listing 3 on page 58. Here is a program written to examine itself. Essentially, it is a memory-dump routine which lists the contents of memory in hexadecimal. As shown, the 15-line program takes 355 bytes and occupies hexadecimal locations 800 through 963 (decimal 2048 through 2499). I have dumped the first and last lines of the program to further demonstrate the storage technique.

I have a reason for explaining the internal program format. One of the useful features of this computer is its ability to function with programs residing solely in EPROM. However,

NO MORE WAITING.

CIS COBOL™ software with FORMS-2™ for Apple II® computers is here. Now. From Micro Focus.

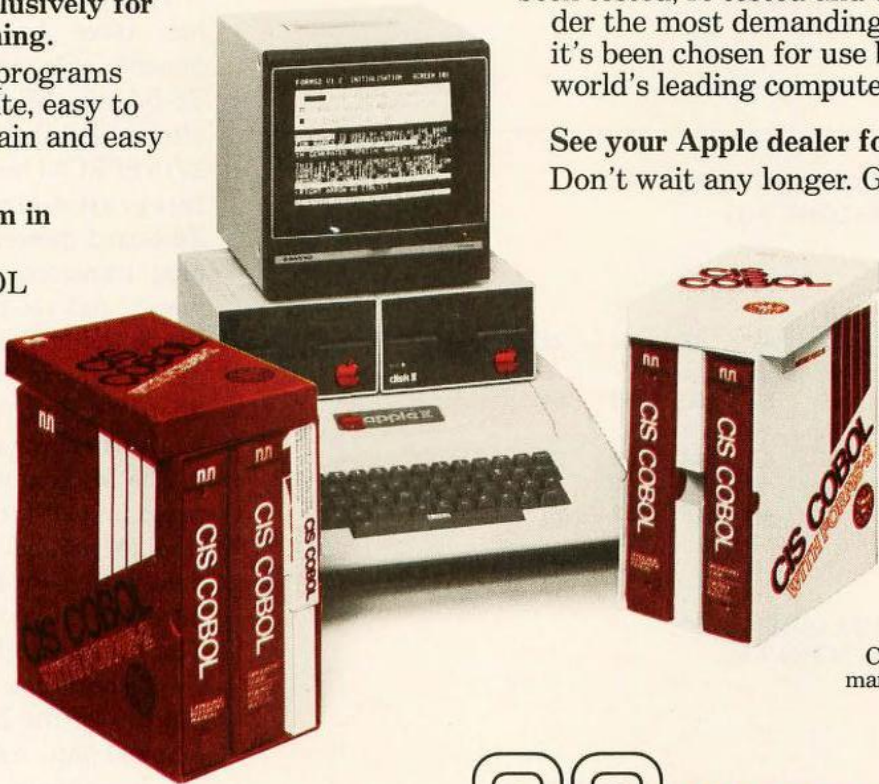
Apple II users . . . the wait is over. And your patience has paid off. Now you can create serious business programs with the world's most popular professional programming language—COBOL. And, run them on the industry standard operating system—CP/M™.

COBOL—the only standard language designed exclusively for business programming.

COBOL gives you programs that are easy to write, easy to read, easy to maintain and easy to learn.

A complete program in 20 minutes.

FORMS-2, a COBOL source-code generator, lets you begin with a blank screen . . . and end with a fully-operational program. FORMS-2 generates standard COBOL code. You can use it as is. Or build on it to create even more sophisticated programs.



Business computer compatibility.

CIS COBOL on Apple II is standard COBOL. (ANSI X3.23-1974) With it, you have the broadest applications flexibility available today. And the most reliable.

Powerful file handling.

You can grow with CIS COBOL. It provides you with all the power of ANSI COBOL file handling. Underlying B-tree technology takes care of large ISAM files . . . fast.

Quick, convenient program development.

With CIS COBOL and FORMS-2, you have your very own low-cost tools that allow you to get work done faster and less expensively than ever before. Features include interactive debug, fast compile-and-go, interactive screen creation and dynamic loading of modules.

GSA approved.

CIS COBOL is certified on Apple II by the General Services Administration. Which means it's been tested, re-tested and tested again . . . all under the most demanding conditions. No wonder it's been chosen for use by over 25 of the world's leading computer manufacturers.

See your Apple dealer for a demonstration.

Don't wait any longer. Get all the details about CIS COBOL and FORMS-2 today. For further information, visit your nearest Apple dealer. Or call us at (408) 248-3982 for a FREE fact-filled brochure.

MICRO FOCUS™
The COBOL Company

Micro Focus, Inc.
1601 Civic Center Dr., Santa Clara, CA 95050
Tel: (408) 248-3982
Telex: 171135 MISSION SNTA

Micro Focus LTD.
58 Acacia Rd., London N.W. 8, ENGLAND
Tel: (01) 722-8843/4/5/6/7
Telex: 28536 MICROF G

Circle 189 on inquiry card.

the EPROMs must be programmed externally. While I will explain how to serially transmit the contents of the program memory to an EPROM programmer, some of you may have only a manual EPROM programmer or one with no communication facility. But if you are willing to spend the time, it is easy to print out the contents of memory and manually load the program into an EPROM device.

Dedicated-Controller Use

The Z8-BASIC Microcomputer can be easily set up for use in intelligent control applications. After being tested and debugged using a terminal, the control program can be written into an EPROM. When power is applied to the microcomputer, execution of the program will begin automatically.

Listing 3: *A program (listing 3a) that examines itself by dumping the contents of memory in printed hexadecimal form. Listing 3b shows the first and last lines of the program as dumped during execution.*

(3a)

```
100 PRINT"ENTER START ADDRESS FOR HEX DUMP ";:INPUT X
102 PRINT"THE LIST IS HOW MANY BYTES LONG ";:INPUT C
103 PRINT:PRINT
105 B=X+8 :A=X+C
107 PRINT"ADDRESS                    DATA":PRINT
110 PRINT HEX (X);"        ";
120 GOSUB 300
130 X=X+1
140 IF X=B THEN GOTO 180
150 GOTO 120
180 IF X>=A THEN 250
200 PRINT:PRINT:B=X+8:GOTO 110
250 PRINT:STOP
300 PRINT HEX (@X);: PRINT"   ";
310 RETURN
:
```

(3b)

```
:RUN
ENTER START ADDRESS FOR HEX DUMP ? 2048
THE LIST IS HOW MANY BYTES LONG ? 30
```

ADDRESS	DATA							
	100		P	R	I	N	T	"
800	0	64	50	52	49	4E	54	22
	E	N	T	E	R	sp	S	T
808	45	4E	54	45	52	20	53	54
	A	R	T	sp	A	D	D	R
810	41	52	54	20	41	44	44	52
	E	S	S	sp	F	O	R	sp
818	45	53	53	20	46	4F	52	20

:

:

```
:RUN
ENTER START ADDRESS FOR HEX DUMP ? 2360
THE LIST IS HOW MANY BYTES LONG ? 45
```

ADDRESS	DATA							
	O	P		300		P	R	I
938	4F	50	0	1	2C	50	52	49
	N	T	sp	H	E	X	sp	(
940	4E	54	20	48	45	58	20	28
	@	X	)	;	:	sp	P	R
948	40	58	29	3B	3A	20	50	52
	I	N	T	"	sp	sp	"	;
950	49	4E	54	22	20	20	22	3B
		310		R	E	T	U	R
958	0	1	36	52	45	54	55	52
	N		65535					
960	4E	0	FF	FF	0	0	0	0

The first application I had for the unit was as a demonstration driver for the Micromouth speech-processor board I presented two months ago in the June issue of BYTE. (See "Build a Low-Cost Speech-Synthesizer Interface," in the June 1981 BYTE, page 46, for a description of this project, which uses National Semiconductor's Digitalker chip set.) It's hard to discuss a synthesized-speech interface without demonstrating it, and I didn't want to carry around my big computer system to control the Micromouth board during the demonstration. Instead, I quickly programmed a Z8-BASIC Microcomputer to perform that task. While I was at it, I set it up to demonstrate itself as well.

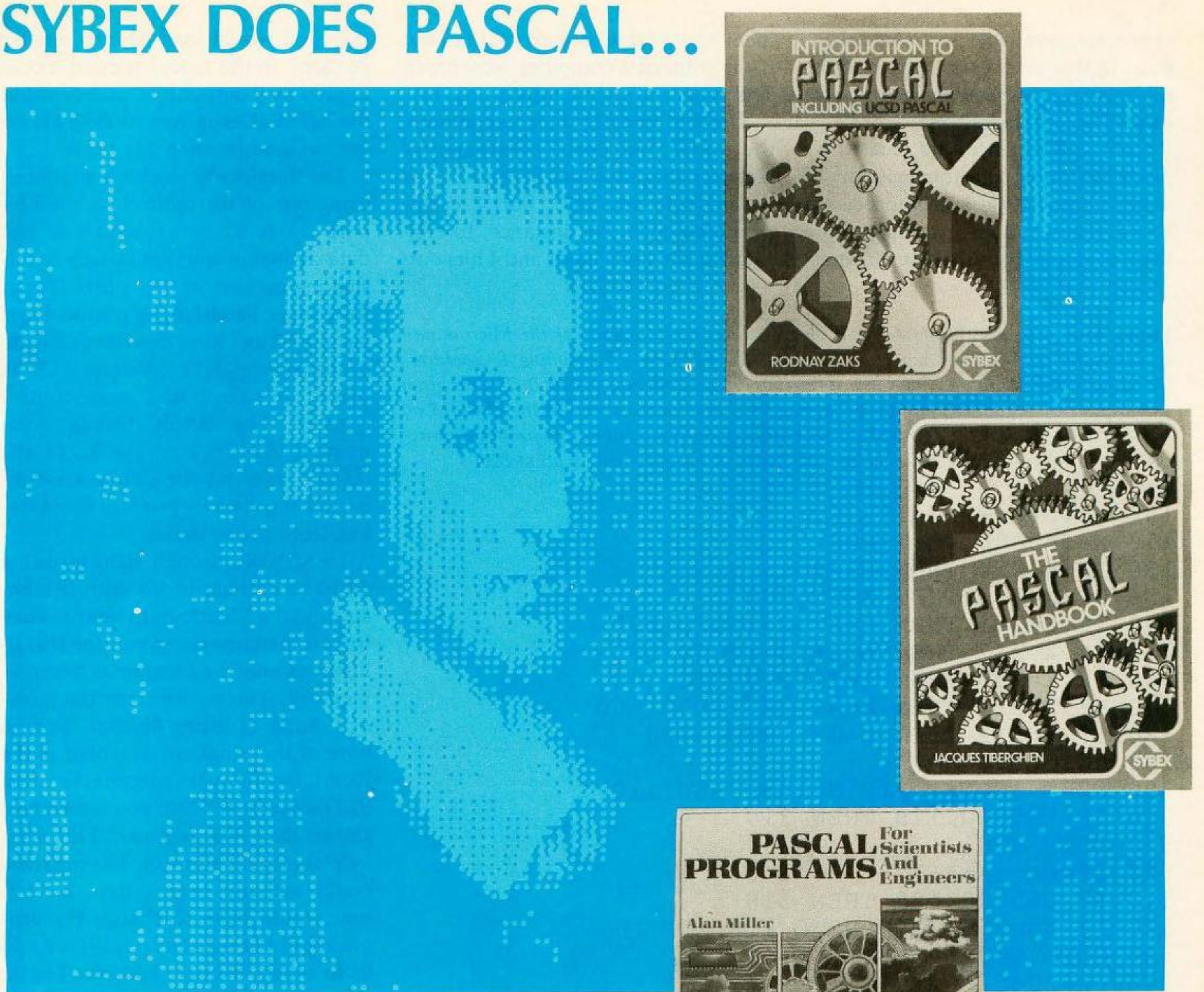
The result (see photo 2 on page 52) has three basic functional components. On top of the box is a Z8-BASIC Microcomputer (hereinafter called the "Z8 board") with a 2716 EPROM installed in the memory integrated-circuit socket, the Z8-board power supply (the wall-plug transformer module is out of view), and six pushbutton switches. Inside the box is a prototype version of the Micromouth speech-processor board (a final-version Micromouth board is shown on the left).

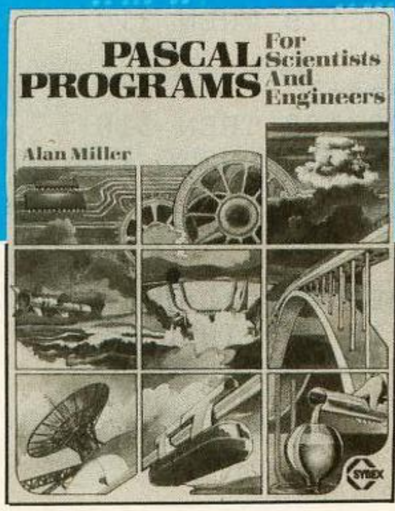
The Micromouth board is jumper-programmed for parallel-port operation (8 parallel bits of data and a data-ready strobe signal) and connected to I/O port 2 on the Z8 board. The Micromouth BUSY line and the six pushbuttons are attached to 7 input bits of the Z8 board's input port mapped into memory-address space at hexadecimal address FFFD (decimal 65533).

The most significant 3 bits of port FFFD are normally reserved for the data-rate-selector switches, but with no serial communication required, the data rate is immaterial and the switches are left in the open position. This makes the 8 bits of port FFFD, which are brought out to the edge

Circle 367 on inquiry card. →

SYBEX DOES PASCAL...





□INTRODUCTION TO PASCAL

by Rodnay Zaks is a simple yet comprehensive guide to standard and UCSD Pascals: step-by-step presentation with exercises for beginners, complex concepts and extensive appendices for experienced programmers. An indispensible book for everyone who wants to learn Pascal programming.
320 pp., 100 illustr., Ref. P310, $14.95

□THE PASCAL HANDBOOK

by Jacques Tiberghien is an easy-to-read, easy-to-use dictionary containing all the features for most existing versions of Pascal (Standard, Jensen-Wirth, OMSI, UCSD, HP1000, Pascal/Z). Over 180 entries, arranged alphabetically; each includes definition, description, syntax diagram, details of implementation, variations and examples. The perfect reference tool for any Pascal user.
500 pp., 150 illustr., Ref. P320, $14.95

□PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS

by Alan Miller is a comprehensive collection of frequently used algorithms for scientific and technical applications programmed in PASCAL. This time saving book includes programs for curve fitting, approximations, random number generation, integrals, statistical techniques and more.
250 pp., 80 illustr., Ref. P340 □$16.95 paper, □$25.00 cloth

MAIL TO:
SYBEX DEPT. B81
2344 SIXTH STREET
BERKELEY, CA 94710
PHONE ORDERS:
INSIDE CA 415/848-8233
TOLL FREE OUTSIDE CA 800-277-2346

NAME ______________________ □SEND ME YOUR FREE CATALOG
ADDRESS ______________________
CITY ______________ STATE ________ ZIP ________
ADD □$1.50/book UPS or □75¢/book 4th class mail or □$8/book overseas airmail
(CA add tax) Total amt. Enclosed ____________ OR CHARGE MY □VISA □MC □AM EX.
CARD NO. ______________________ EXP. DATE ________
SIGNATURE ______________________

connector, available for external inputs. In this case, pressing one of the six pushbuttons selects one of six canned speech sequences.

Coherent sentences are created by properly timing the transmission of word codes to the speech-processor board. This requires nothing more than a single handshaking arrangement and a table-lookup routine (but try it without a computer sometime). The program is shown in listing 4a.

The first thing to do is to configure the port-2 and port-3 mode-control registers (hexadecimal F6 and F7, or decimal 246 and 247). Port 2 is bit-programmable. For instance, to configure it for 4 bits input and 4 bits output, you would load F0 into register F6 (246). In this case, I wanted it configured as 8 output bits, so I typed in the BASIC/Debug command @246=0 (set decimal location 246 to 0).

The data-ready strobe is produced using one of the options on the Z8's port 3. A Z8 microcomputer has data-available and input-ready handshaking on each of its 4 ports. To set the proper handshaking protocol and use port 2 as I have described, a code of hexadecimal 71 (decimal 113) is placed into the port-2 mode-control register. The BASIC/Debug command is @247= 113. The RDY2 and $\overline{DAV2}$ lines on the Z8671 are connected together to produce the data-available strobe signal.

Lines 1000 through 1030 in listing 4a have nothing to do with demonstrating the Micromouth board. They form a memory-dump routine that illustrates how the program is stored in memory. You notice from the memory dump of listing 4b that the first byte of the program, as stored in the ROM, begins at hexadecimal location 820 (actually at 1020, you remember) rather than 800 as usual. This is to help automatic start-up. The program could actually begin anyplace, but you would have to change the program-pointer registers (registers 8 and 9) to reflect the new address. The 32 bytes between 800 and 820 are reserved for vectored addresses to optional user-supplied I/O drivers and interrupt routines.

Listing 4: *A program (listing 4a) that demonstrates the functions of the Micromouth speech synthesizer, operating from a type-2716 EPROM. The simple I/O-address decoding of the Z8 board allows use of the round-figure address of 65000. The program uses a table of vocabulary pointers that has been previously stored in the EPROM by hand. Listing 4b shows a dump of the memory region occupied by the program, proving that storage of the BASIC source code starts at hexadecimal location 820.*

(4a)

```
100 @246=0:@247=113
110 X=@65000 :A=%1400
120 IF X=254 THEN @2=0
130 IF X=253 THEN GOTO 500
140 IF X=251 THEN A=A+32 :GOTO 500
150 IF X=247 THEN A=A+64 :GOTO 500
160 IF X=239 THEN A=A+96 :GOTO 500
170 IF X=223 THEN A=A+128 :GOTO 500
180 IF X=222 THEN N=0 :GOTO  300
200 GOTO 110
300 @2=N :N=N+1 :IF N=143 THEN 110
310 IF @65000<129 THEN 310
320 GOTO 300
500 @2=@A :A=A+1
510 IF @65000<129 THEN 510
520 IF @A=255 THEN GOTO 110
530 GOTO 500
1000 Q=2048
1005 W=0
1010 PRINT HEX(@Q),:Q=Q+1
1015 W=W+1 :IF W=8 THEN PRINT" ":GOTO 1005
1020 IF Q=4095 THEN STOP
1030 GOTO 1010
:
```

(4b)

```
:goto 1000
FF      FF      FF      FF      FF      FF      FF      FF
FF      FF      FF      FF      FF      FF      FF      FF
FF      FF      FF      FF      FF      FF      FF      FF
FF      FF      FF      FF      FF      FF      FF      FF
0       64      40      32      34      36      3D      30
3A      40      32      34      37      3D      31      31
33      0       0       6E      58      3D      40      36
35      30      30      30      20      3A      41      3D
25      31      34      30      30      0       0       78
49      46      20      58      3D      32      35      34
20      54      48      45      4E      20
0! AT 1015
:
```

Programming the EPROM

The first EPROM-based program I ran on the Z8-BASIC Microcomputer was manually loaded. I simply

The System/48 is the outstanding office automation computing system for the 80's . . . it's so productive we call it MAGIC©. Look at these features:

- One to eight interactive users per node
- Data management system
- Report generator
- Query processor
- Screen format generator
- Automatic interface code generator
- Up to ½ Mbytes of user memory
- Winchester disk with 18-million bytes (formatted capacity)
- 15-minute mean-time-to-repair
- Built-in protection from line-voltage spikes, noise, and brownouts

And, it features MAGIC©, the Operating System that gets things done faster than you can say abracadabra because of its multi-keyed Indexed-Sequential Access Method and flexible file-organization. MAGIC© also offers high security, with password protection. MAGIC© supports global or local printers for as many users as desired.

MAGIC© also includes DataMagic II — TEI's red-hot database manager. DataMagic II has even more tricks up its sleeve — like automatic or manual record-lock protection and automatic transaction backout to protect the database. And it runs application software written for CP/M 2.X.

Take a MAGIC© leap into the future!

Arrange to attend one of our regularly scheduled System/48 workshops (RSVP).

OEM and Dealer Inquiries Invited

TEi INC. | System/48®

5075 S. LOOP EAST, HOUSTON, TX. 77033
(713) 738-2300 TWX. 910-881-3639

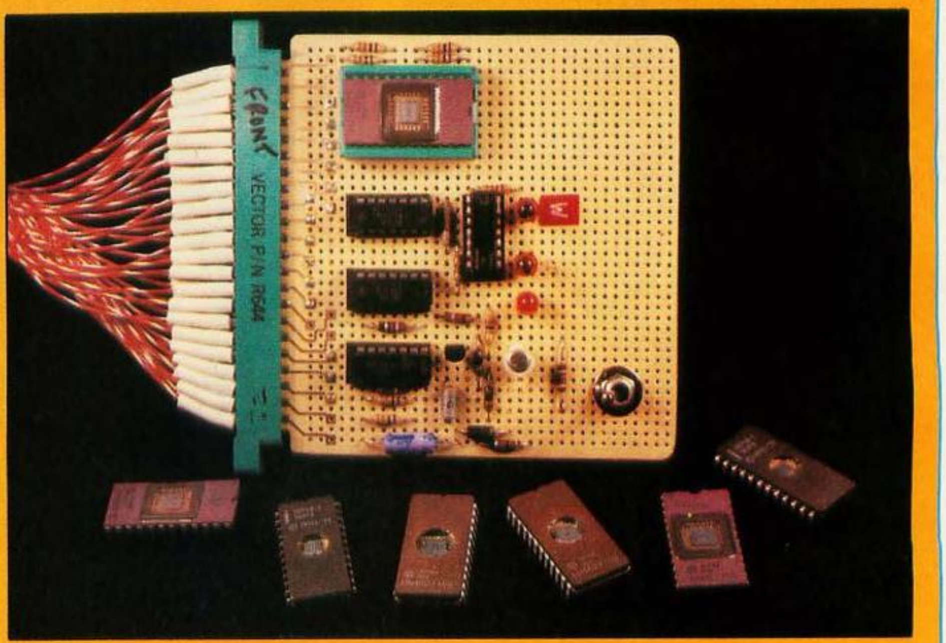


Photo 3: *Type-2716 EPROM programmer, adapted from "Program Your Next EROM in BASIC" (March 1978 BYTE, page 84). The circuit, which is driven through parallel ports, programs a 2716 in about 2½ minutes and is controlled by a BASIC program.*

printed out the contents of the Z6132 memory using the program of listing 3 and entered the values by hand into the EPROM programmer. This is fine once or twice, but you certainly wouldn't want to make a habit of it. Fortunately, there are better alternatives if you have the equipment.

Many EPROM programmers are peripheral devices on larger computer systems. In such cases, it is possible to take advantage of the systems' capabilities by downloading the Z8 program directly to the programmer.

The programmer shown in photo 3 is a revised version of the unit I described in a previous article, "Program Your Next EROM in BASIC" (March 1978 BYTE, page 84). It was designed for type-2708 EPROMs, but I have since modified it to program 2716s instead. All I had to do was lengthen the programming pulse to 50 ms and redefine the connections to four pins on the EPROM socket. It still is controlled by a BASIC program and takes less than 2½ minutes to program a type-2716 EPROM device. Refer to the original article for the basic design.

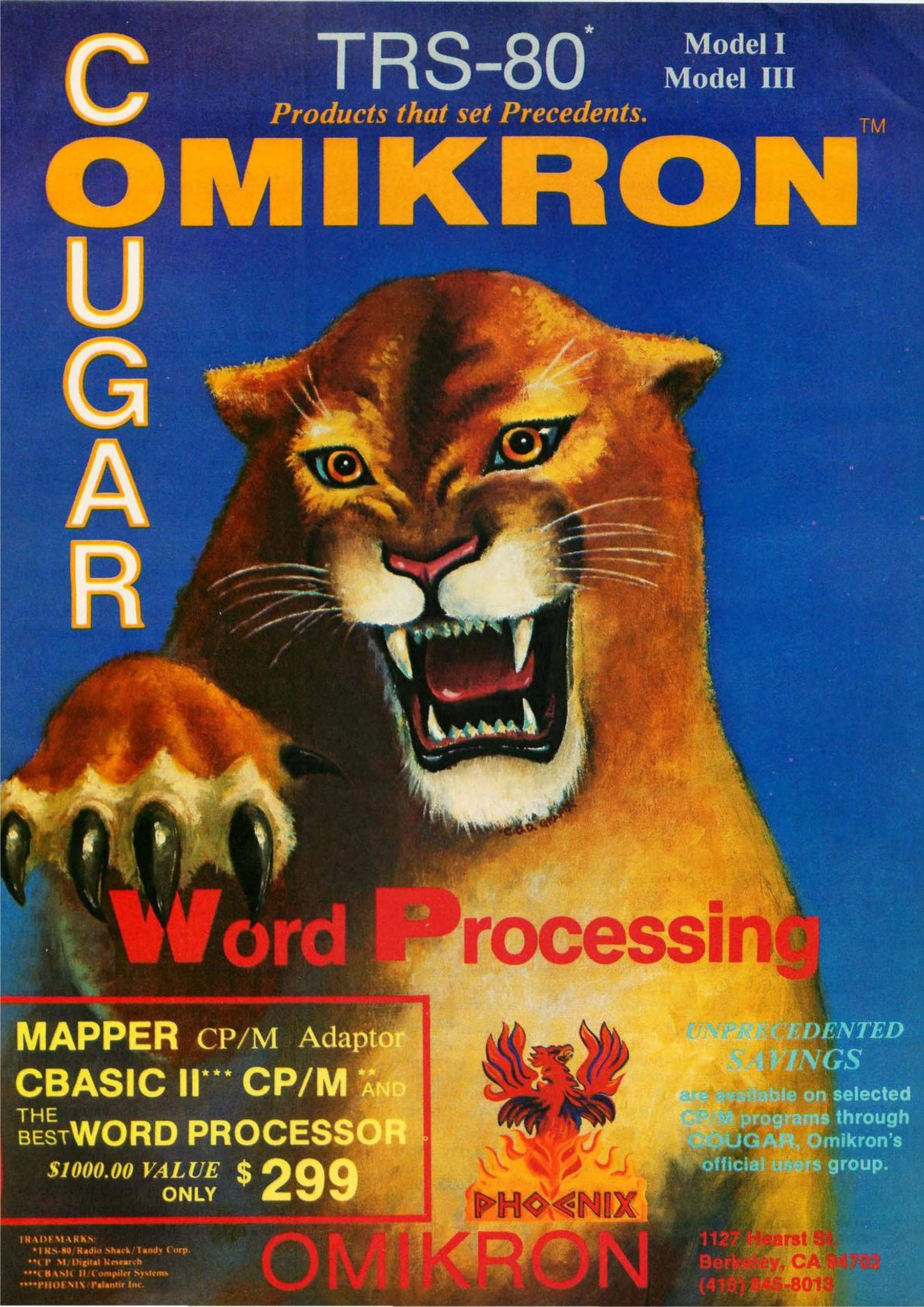
Normally, the LIST function or memory-dump routine cannot be used to transmit data to the EPROM programmer because the listing is filled with extraneous spaces and carriage returns. It is necessary to write a program that transmits the contents of memory without the extra characters required for display formatting. The only data received by the EPROM programmer should be the object code to load into the EPROM.

In writing this program we can take advantage of the Z8's capability of executing machine-language programs directly through the USR and GO@ commands. The serial-input and serial-output subroutines in the BASIC/Debug ROM can be executed independently using these commands. The serial-input driver starts at hexadecimal location 54, and the serial-output driver starts at hexadecimal location 61. Transmitting a single character is simply done by the BASIC statement

```
GO@ %61,C
```

where C contains the value to be transmitted. A serial character can be received by

```
C=USR (%54)
```



COUGAR
TRS-80*
Model I
Model III
Products that set Precedents.
OMIKRON™
C.O.D. GRAPH
Word Processing
MAPPER CP/M Adaptor
CBASIC II*** CP/M** AND
THE BEST WORD PROCESSOR
$1000.00 VALUE ONLY $299
PHOENIX
UNPRECEDENTED SAVINGS
are available on selected CP/M programs through COUGAR, Omikron's official users group.
TRADEMARKS:
*TRS-80/Radio Shack/Tandy Corp.
**CP/M/Digital Research
***CBASIC II/Compiler Systems
****PHOENIX/Palantir Inc.
OMIKRON
1127 Hearst St.
Berkeley, CA 94702
(415) 845-8013

Listing 5: *BASIC statements that print out the entire contents of the 4 K bytes of user memory, for use with a communicating EPROM programmer.*

```
1000 X=%800 :REM BEGINNING OF
     USER MEMORY
1010 GO@ %61,@X :REM TRANSMIT
     CONTENTS OF LOCATION X
1020 X=X+1 :IF X=%1801 THEN
     STOP
1030 GOTO 1010
```

Listing 6: *A simple BASIC program segment to demonstrate the concept of the "black box" method of modifying data being transmitted through the Z8-BASIC Microcomputer.*

```
100 @246=0:@247=113 :REM SET PORT
    2 TO BE OUTPUT
110 @2=X :REM X EQUALS THE DATA
    TO BE TRANSMITTED
```

where the variable C returns the value of the received data.

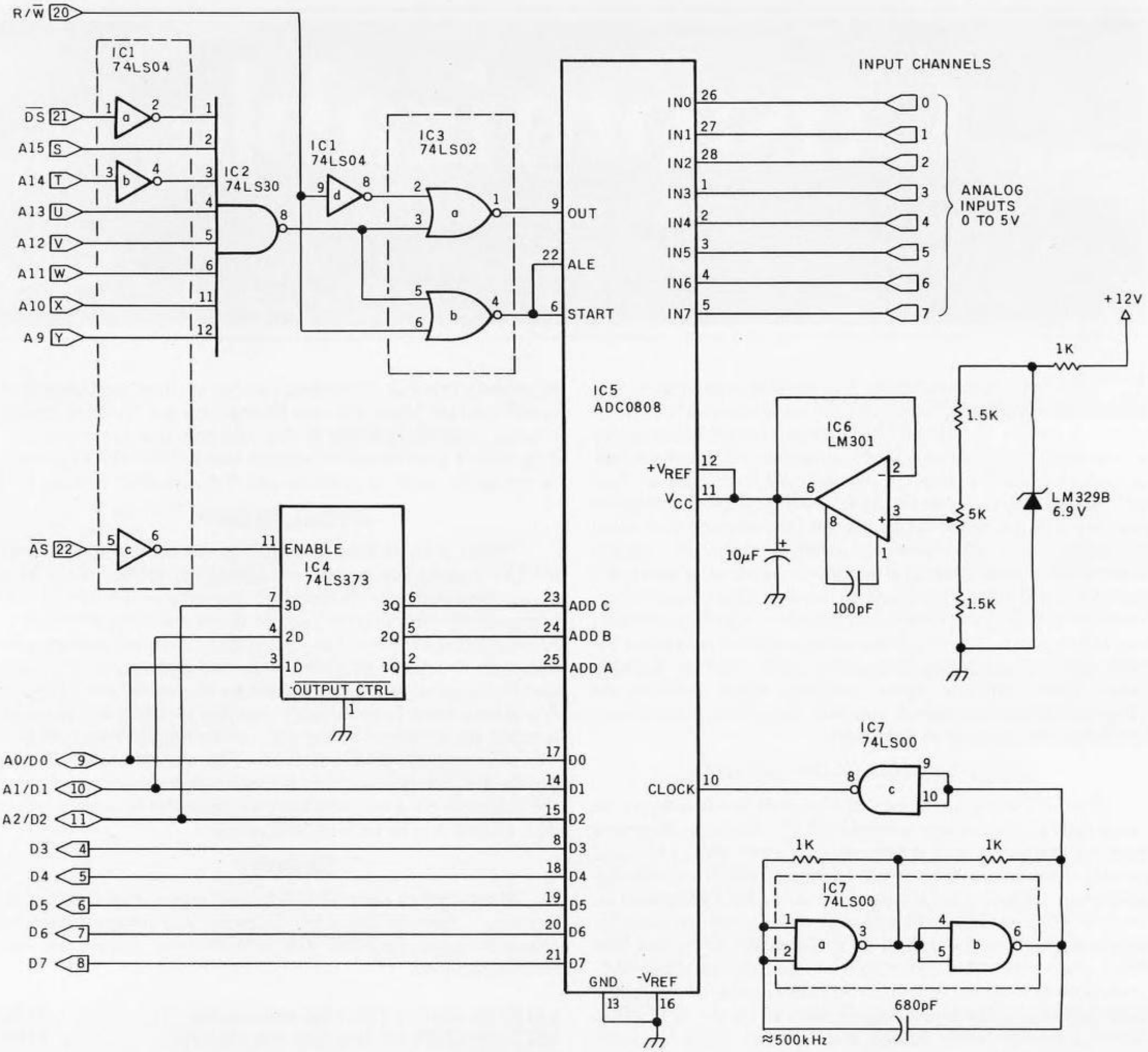
To dump the entire contents of the Z6132 memory to the programmer, the statements in listing 5 should be included at the end of your program.

Execution begins when you type GOTO 1000 as an immediate-mode command and ends when all 4 K bytes have been dumped. The transmission rate (110 to 9600 bps) is that selected on the data-rate-selector switches.

Conceivably, this technique could also be used to create a cassette-storage capability for the Z8 board. In theory, a 3- or 4-line BASIC program can be entered in high memory (you can set the pointer to put the program there) to read in serial data and load it in lower memory. Changing the program pointer back to hexadecimal 800 allows the newly loaded program to be executed. Since the Z8-BASIC Microcomputer already has a serial I/O port, any FSK (frequency-shift keyed) modem and cassette-tape recorder can be used for cassette data storage.

I/O for Data Acquisition

Data acquisition for process control is the most likely application for the Z8-BASIC Microcomputer. Low-

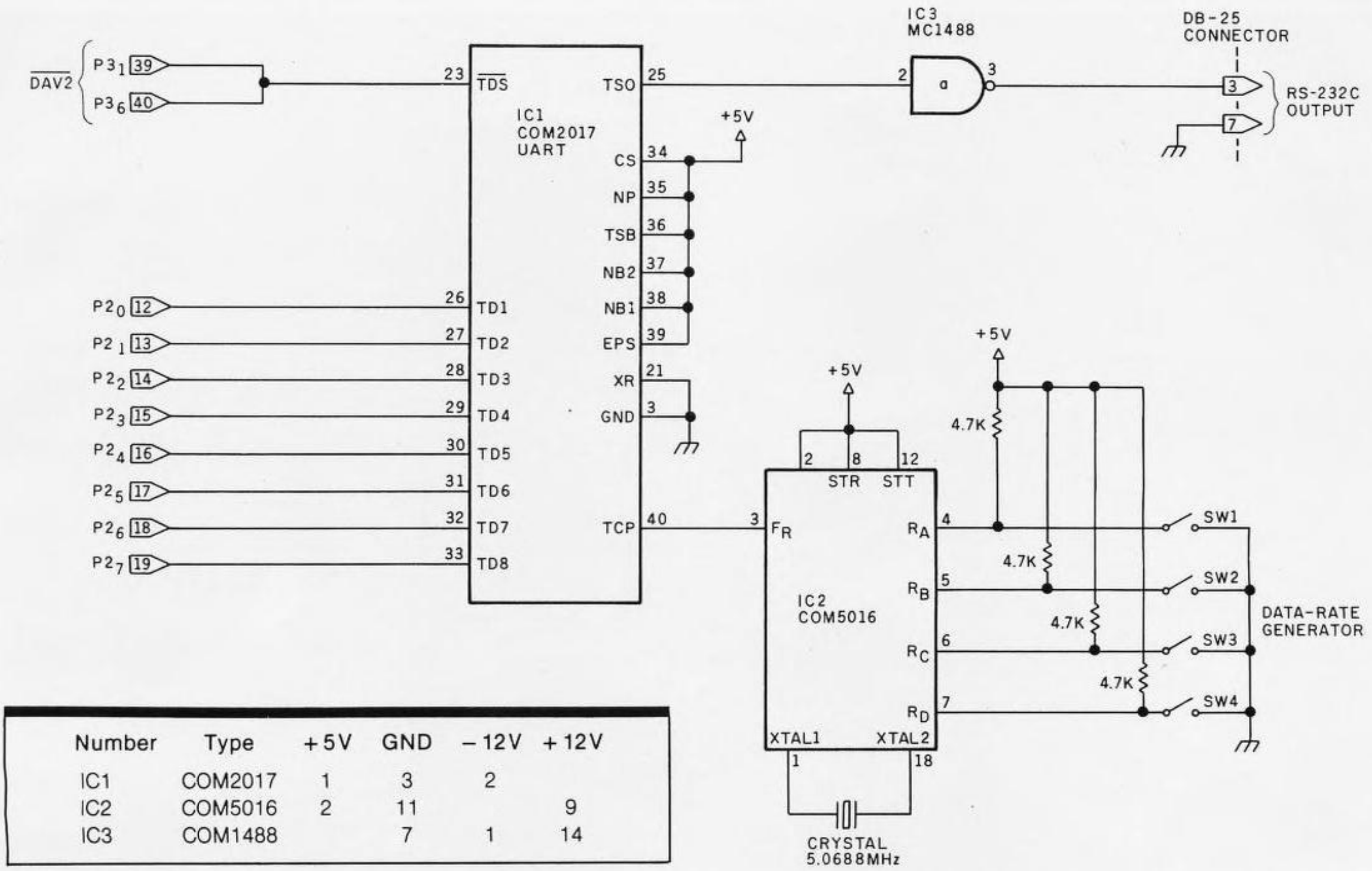


Number	Type	+5V	GND	+12V
IC1	74LS04	14	7	
IC2	74LS30	14	7	
IC3	74LS02	14	7	
IC4	74LS373	20	10	
IC5	ADC0808	see schematic diagram		
IC6	LM301		4	7
IC7	74LS00	14	7	

Figure 2: *Schematic diagram of an A/D converter. This 8-bit, eight-channel unit has a unipolar input range of 0 to +5 V, with the eight output channels addressed as I/O ports mapped into memory-address space at hexadecimal addresses BF00 through BF07.*

cost distributed control is practical, substituting for central control performed by a large computer system. Analog and digital sensors can be read by a Z8-BASIC Microcomputer, which then can digest the data and reduce the amount of information (experiment results or control parameters) stored or transmitted to a central point. Control decisions can be made by the Z8-BASIC Microcomputer at the process locality.

The Z8 board can be used for analog data acquisition, perhaps using an A/D (analog-to-digital) converter such as that shown in figure 2. This 8-bit, eight-channel A/D converter has a unipolar input range of 0 to +5 V (although the A/D integrated circuit can be wired for bipolar operation), with the eight output channels addressed as I/O ports mapped into memory-address space at hexadecimal addresses BF00 through BF07 (decimal 48896 through 48903). When the Z8671 performs an output operation to the channel address, the channel is initialized for acquiring data, while data is read from the channel when the Z8671 performs



Number	Type	+5V	GND	−12V	+12V
IC1	COM2017	1	3	2	
IC2	COM5016	2	11		9
IC3	COM1488		7	1	14

Figure 3: *Schematic diagram of an RS-232C serial output port for the "black box" communication application of the Z8-BASIC Microcomputer. The Z8671 must be configured by software to provide the proper signals: one such signal, DAV2, is derived from two bits of I/O port 3 on the Z8671. The pin numbers shown in the schematic diagram for* $P3_1$ *and* $P3_6$ *are pins on the Z8671 device itself, not pins or sections on the card-edge connector, as are* $P2_0$ *through* $P2_7$.

an input operation on the channel's address.

Intelligent Communication

Another possible use for the Z8-BASIC Microcomputer is as an intelligent "black box" for performing predetermined modification on data being transmitted over a serial communication line. The black box has two DB-25 RS-232C connectors, one for receiving data and the other for retransmitting it. The intelligence of the Z8-BASIC Microcomputer, acting as the black box, can perform practically any type of filtering, condensing, or translating of the data going through.

Perhaps you have an application where continuous raw data is transmitted, but you would rather just keep a running average or flag deviations from preset limits at the central monitoring point rather than contend with everything. The Z8 board can be programmed to digest all the raw data coming down the line and pass

NEW!
"WORDSTAR™" goes APPLE!!

It's amazing! It's colossal! Awesome Apple at last joins forces with world-famous WordStar. And together they easily conquer the most beastly word processing! If you're an Apple owner, this is one double feature you won't want to miss!

WordStar, after all, is the most powerful and versatile word processing software ever developed for microcomputers. And now it's available for Apple—thanks to the SoftCard from Microsoft, a Z-80-based circuit card that converts Apple to CP/M. And thanks to a new edition of WordStar formatted especially for Apple.

You'll be glad to know our Apple edition contains all the features that have made WordStar the top banana in word processing:

What you see is what you get. With WordStar the screen shows exactly what will get printed—including all format variables like page breaks, justification, and hyphenation. As soon as you type a change, it's onscreen.

User friendly? Our menus give you so much help that most of the time you can leave your manual sitting on the shelf. Choose from four levels of help according to the job and your own experience.

WordStar has more talent than you may ever need. Its flexibility is simply awesome. From text typing to the most complex word processing you can imagine—WordStar lets you sail through it all with ease and efficiency.

In fact, WordStar is so powerful and so friendly, we challenge comparison—feature for feature—against dedicated word processing systems costing thousands more than a WordStar-based system.

So, if you love your Apple but wish it could do more, that's where we come in. First, with WordStar. And next—since we don't monkey around when it comes to software—with Apple editions of our whole software family, all designed to work together:

SpellStar™, a new WordStar option, will find your spelling and typing mistakes for you. **MailMerge**™, another WordStar option, makes it easy to combine files to produce personalized form letters. **SuperSort**™ takes on the biggest sorting, merging, and selecting jobs. And **DataStar**™ handles data entry, retrieval, and update with tremendous power and precision.

Innovative excellence—that's what has made WordStar such a smash among micro users. And that's what you'll find in MicroPro's entire family of business software—now available in Apple editions.

Want more information about our Apple software? Clip the coupon. Or, if you've really gone ape by now, just phone us for the name of your nearest dealer.

Circle 230 on inquiry card.

MicroPro™
INTERNATIONAL CORPORATION
1299 Fourth Street
San Rafael, California 94901
(415) 457-8990
Telex 340-388

Available in 40 & 80 column screen versions; and 13 & 16 sector media.

YES,

I've gone ape!
Please rush me information on MicroPro's family of business software solutions, and the name of my local MicroPro dealer.

Name ____________
Title ____________
Company ____________
Address ____________
City ____________ State ______ Zip ______

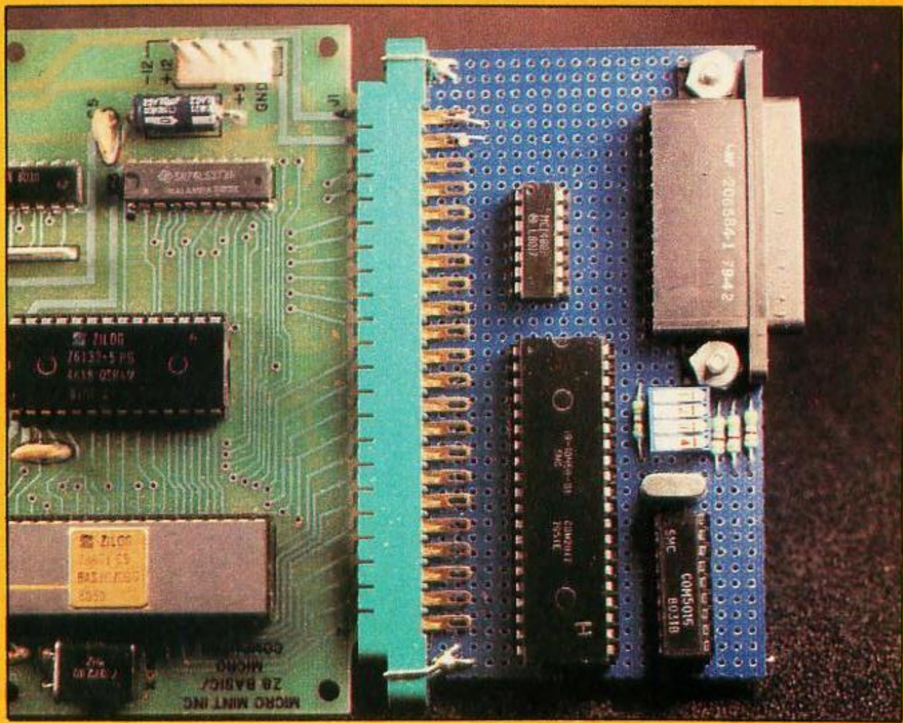


Photo 4: *A three-integrated-circuit hardwired serial output port for the Z8-BASIC Microcomputer. Connected to port 2, any program data sent to register 2 will be transmitted serially at the data rate selected on the four-position DIP switch (between 50 to 19200 bps). The Z8 board, configured with two serial ports, is used to process raw data moving through it. Data is received on one side, digested, and retransmitted in some more meaningful form from the other port. Such a configuration could also be used to connect two peripheral devices that have radically different data rates.*

on only what's pertinent.

Another such black-box application is to use the Z8 board as a printer buffer. Photo 4 shows the interface hardware of one specific application, which I used to attach a high-speed computer to a very slow printer. The host computer transmitted data to the Z8 board at 4800 bps. Since the receiving serial port used had to be bidirectional to handshake with the host computer, I added another serial output to the Z8 board for transmitting characters to the printer. Only three integrated circuits were required to add a serial output port. A schematic diagram is shown in figure 3 on page 67. The UART (universal asynchronous receiver/transmitter, shown as IC1) is driven directly from port 2 on the Z8 board (port 2 could also be used to directly drive a parallel-interface printer), and IC2 supplies the clock signal for the desired data rate. Of course, the UART could have been attached to the data and address buses directly, but this was easier.

Transmitting a character out of this serial port requires setting the port-2 and port-3 mode-control registers as before. After that, any character sent to port 2 will be serially transmitted. The minimum program to perform this is shown in listing 6 on page 64. This circuit can also be used for downloading programs to the EPROM programmer.

In Conclusion

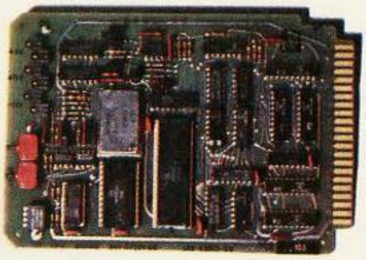
It is impossible to describe the full potential of the Z8-BASIC Microcomputer in so few pages. For this reason, considerable effort has been taken to fully document its characteristics. I have merely tried to given an introduction here.

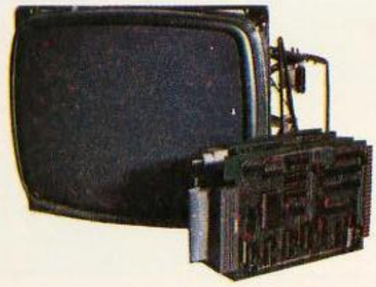
I intend to use the Z8-BASIC Microcomputer in future projects. I am interested in any applications you might have, so let me know about them, and we can gain experience together.

We know just how good G&L Microcomputers are.

Because we use them in our own PC's and CNC's.

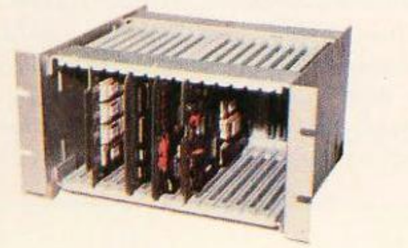
And now we can fill all your needs with reliable industrial grade modules.





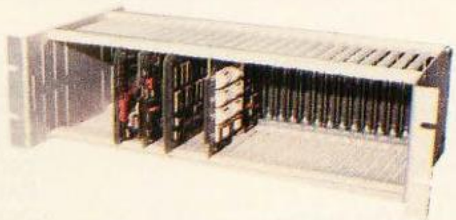
GL 868 MODULES

- CPU-FPA PROCESSOR
- RAM-CMOS-EPROM
- ANALOG MODULES
- VIDEO MODULES
- SERVO MODULES



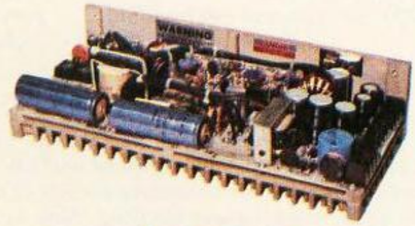
NESTS-POWER SUPPLIES

- HALF NEST
- FULL NEST
- MASTER RACK
- POWER SUPPLIES



PERIPHERALS

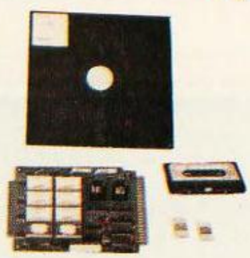
SOFTWARE PACKAGES



DEVELOPMENT SYSTEMS

RELIABILITY

- 130°F AMBIENT RATED
- 95% RELATIVE HUMIDITY
- PRE-TESTED CHIPS WITH 125°C, 168 HOUR BURN-IN
- MODULE BURN-IN AND TEST AT 130°F FOR 40 HOURS



GET THE FACTS
Ask for the fact-packed GL868 booklet.
Giddings & Lewis Electronics Co.,
P.O. Box 1658, 666 S. Military Road,
Fond du Lac, WI 54935, U.S.A. Telephone
(414) 921-9400.

Electronics Division
Giddings & Lewis

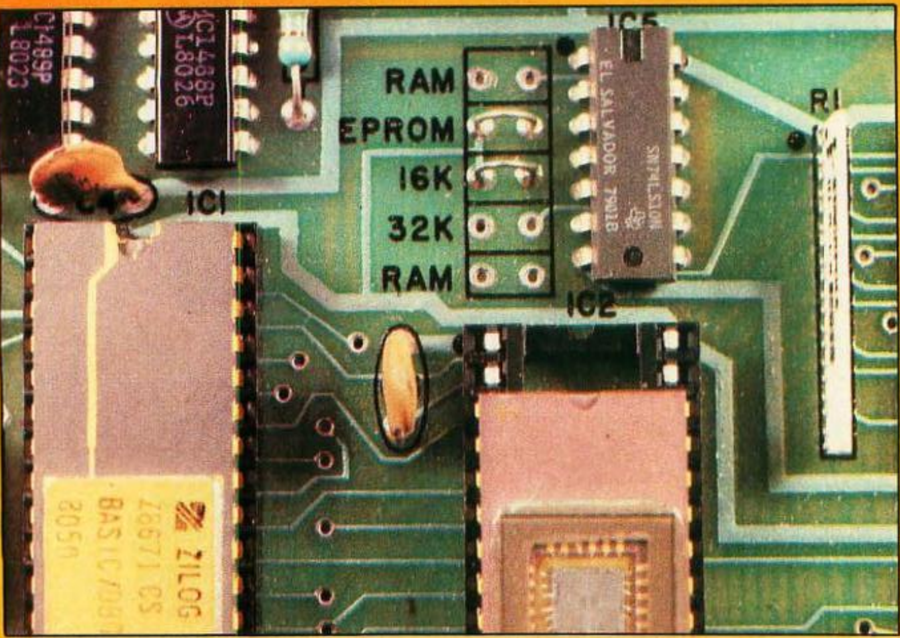


Photo 5: *When the Z8-BASIC Microcomputer is used with a ROM-resident program, the two jumpers used with the Z6132 are removed, and the EPROM jumper is installed instead. When using a type-2716 16 K-bit (2 K-byte) EPROM device, the "16 K" jumper is installed. If a type-2732 32 K-bit (4 K-byte) EPROM is used instead, the "32 K" jumper is installed. The EPROM is inserted in the lower 24 pins of the 28-pin Z6132 socket (IC2) as shown.*

Next Month: *Build a phonetic voice synthesizer based on the Votrax SC-01 synthesizer chip.* ■

Special thanks to Steve Walters and Peter Brown of Zilog Inc for their aid in producing these articles.

Editor's Note: *Steve often refers to previous Circuit Cellar articles as reference material for the articles he presents each month. These articles are available in reprint books from BYTE Books, 70 Main St, Peterborough NH 03458.* Ciarcia's Circuit Cellar *covers articles appearing in BYTE from September 1977 through November 1978.* Ciarcia's Circuit Cellar, Volume II *presents articles from December 1978 through June 1980.*

To receive a complete list of Ciarcia Circuit Cellar kits available from The MicroMint, circle 100 on the inquiry card.

The following items are available postpaid in the United States from:
The MicroMint Inc
917 Midway
Woodmere NY 11598
Telephone:
(800) 645-3479 (for orders)
(516) 374-6793 (for technical information)

Z8-BASIC Microcomputer
(Documentation includes:
Z8 Technical Manual
Z8 Product Specification
Z6132 Product Specification
BASIC/Debug Manual
Z8-BASIC Microcomputer Construction/Operator's Manual)
Assembled and tested....$170
Kit....$140

Z8-BASIC Microcomputer power supply
(Size: 2⅛ by 4½ inches)
Provides: +5 V, 300 mA
+12 V, 50 mA
−12 V, 50 mA
Assembled and tested....$35

These prices are in effect until September 15, 1981; call for prices after that date.
All printed-circuit boards are solder-masked and silk-screened. The documentation supplied with the Z8 board includes approximately 200 pages of materials. It is available separately for $25. This charge will be credited toward any subsequent purchase of the Z8 board.
Please include $2 for shipping and handling. New York residents please include 7% sales tax.