LIFELINES®

Published monthly by Lifelines Publishing Corporation Volume 1 No. 8 January 1981

Editor-in-Chief: Harris Landgarten
Managing Editor: Jane Mellin
Contributing Editor: Ward Christensen
Production Assistant: Mary Anna Feczo
Offices: 1651 Third Avenue, New York, N.Y. 10028
Telephone: (212) 722-1700

Subscription Information:

Domestic: (including Canada and Mexico):
$18 for twelve issues
$2.50 per single issue

Foreign: $40 for twelve issues
$3.60 per single issue

Order from Lifelines Publishing Corporation at the address above. All orders must be prepaid. Foreign checks must be in U.S. dollars, drawn on a U.S. bank. Checks, money orders, MasterCard and VISA are acceptable. Please make all checks payable to Lifelines Publishing Corporation.

CONTENTS

Editorial Comments

As 1981 begins, it is appropriate to look back at Lifelines' record for the past year. Certainly, it was a very successful year. In just seven months, we have developed a paid readership well in excess of three thousand, a readership comprised of the most microcomputer-literate people in the world. Ever since my first involvement with micros in 1975, I had acutely felt the dearth of software support. As I grew from hobbyist to professional, the problem of keeping up with the latest information on a timely basis also grew. Even if the behemoth computer magazines wish to publish meaningful articles and reviews, they cannot maintain timeliness because of their enormous lead times. When was the last time you saw the current version of a software product reviewed in Byte? I am glad to see that I am not alone in this quest for accurate, up-to-date information. With your continuing support, Lifelines will grow to meet the needs of this expanding industry.

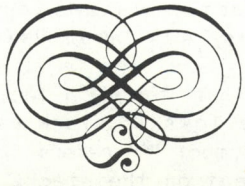
In our first seven issues, we have supported--via a version list, update notices and bug notices--the products sold by Lifeboat Associates. While this encompasses many products, it by no means represents the totality of serious 8080/Z80 software. Therefore, in future issues, we will endeavor to better cover the entire market. To this end, I ask you to tell us what products you wish to see added to our monthly features. If you want to see Lifelines report on Electric Pencil, Pickles and Trout CP/M, Visicalc or any other worthy software product, just drop us a line and we will contact the author. Furthermore, software authors who wish to keep their customers informed of the latest developments in their programs are urged to contact us.

In the past few months, there has been a growing amount of interest in more advanced operating systems for microcomputers. Oasis, MP/M, CP/NET and the new Unix-like systems are the ones most often mentioned. This is the forefront of microcomputer technology and many questions have to be answered. Among the most important ones are "which ones are real?", "who needs them?", "how do I get them 'up'" and "what is available to run under them?". Lifelines will publish columns covering these alternative operating systems, if enough interest is expressed.

The CP/M Users Group has been growing at a fantastic rate. It's no wonder since it is the biggest bargain around. Unfortunately, the number of disk formats in use seems to be growing as fast as the Users Group. Recently the entire CPMUG library became available on North Star format disks. Because of the wonderful response, the Users Group is contemplating a major expansion in the number of disk formats supported. Superbrain, Heath and Micropolis are among those being considered. If the interest is great enough these and others will be supported. Let us know what you want.

1981 will be the year that Lifelines expands to encompass the information needs of a growing number of microcomputer users. Watch our pages in the coming months for some of the things I've mentioned along with some possibly astounding surprises. Oh, I almost forgot; Happy New Year to you, your family and your computer.

Harris Landgarten



Assembly Language Development Systems

SID has every one of the DDT commands as a subset of SID's commands. However, each is enhanced beyond the DDT capabilities. Also, there are commands which are unique to SID. They are:

```
C  Call program
H  Hex arithmetic, symbol dump
P  Pass counter
```

(Note: In DDT there is an undocumented "H" command.-Editor)

That may not sound like much, but taken in combination with the significant extensions to the basic DDT commands, SID becomes a truly fun and productive debugging tool.

Let's look at the commands in detail. I'll comment on the improvements in SID over DDT, and where pertinent, the deficiencies of DDT. It is common to all commands which take an address that when using a symbol table with SID you may use the symbol instead of its hex value. To do so, use "." followed by the symbol. This tells SID to look up the symbol in its symbol table and substitute the value of the symbol. For example, if you want to make a patch to a routine at label "TEST", you don't have to go to a listing to find out where TEST is; just type:

```
a.test
```

and SID will look up the label TEST and substitute its value.

As if that wasn't enough, you can also do indirect memory addressing:

```
D.TEST
```

dumps starting at location TEST.

```
D@POINTER
```

dumps memory, using the value stored into pointer. Note how this differs from D.POINTER, which dumps POINTER itself. Skip the next paragraph if you understand.

Suppose POINTER is a DW, and has the value of BUFFER in it.

Programmatically:

```
180  POINTER   DW   BUFFER
                ...
19C  BUFFER    DS   200
```

If you type

```
D.POINTER
```

you will be dumping memory starting at 180. If you type

```
D@POINTER
```

you will be dumping memory starting at 19C, the value CONTAINED in POINTER.

Similarly, the notation:

```
=SYMBOL
```

refers indirectly to the BYTE stored at SYMBOL. For example, if in the above programming example BUFFER is 128 bytes long, and DISP is a DB which contains a displacement into BUFFER,

```
D.BUFFER+=DISP
```

will dump BUFFER, starting at the displacement in DISP; i.e. the byte IN DISP is loaded, and added to the value OF BUFFER (its address). The dump is begun there. Whew!

And still SID adds more ways to just reference data; DDT only allows HEX data. SID adds to that with symbolic and indirect symbolic, as we have seen. It also adds decimal and character data types to its repertoire. For example, 10 means hex 10 (as in DDT) but #10 means decimal 10. Character values appear in quotes, e.g. 'a' or 'XX'.

One "last" new data format which SID allows is stack reference. The symbol "^" refers to the top element on the stack. For example, suppose you were tracing and were in the middle of a subroutine. If you wanted to "go" but breakpoint at the return address, you would type:

```
G,^
```

The "," indicates an omitted starting address, i.e. continue with present PC. The "^" indicates the need to set a breakpoint at the address on the top of the stack.

If the subroutine had pushed a register, then the return address would be the second element on the stack, so you just use as many consecutive "^" characters as necessary to represent that depth into the stack. For example, to go and breakpoint at the return address of a subroutine, which has done one PUSH, type:

```
G,^^
```

(A)ssemble

The (A)ssemble command under SID has two new "convenience" features. Did you ever mistype a command in DDT, and have to rekey in the Axxxx command? Well, with SID, it remembers where it left off, so just an A alone will continue assembling at the proper address. Also, the command "-A" says to delete the assemble/disassemble (A and L) modules, and the symbol table. You might occasionally use this when you want the maximum memory available. I have had no need for it, so far.

(C)all

The (C)all program command is new to SID and presents a simple way to call a subroutine of SID. Optionally, you may pass parameters in BC and DE to the called routine. Do not confuse

by Ward Christensen

this with calling a subroutine in the program you are debugging. Suppose you have a routine which you have patched into "high" memory, and want to execute it without changing the PC or other registers currently used by SID. Just Cnnnn, to call it. NOTE that the C command is most frequently used with the SID utilities, which may be discussed at a later date. (I haven't yet used them myself.)

(D)ump

The (D)ump memory command now allows a format of "+nn" for its second operand. This assumes the value of the first operand, plus nn.

For example, to dump the first 16 (10H) bytes in BUFFER, you can just:

```
D.BUFFER,+10
```

instead of having to say:

```
D.BUFFER,.BUFFER+10
```

Also, if you have data in 16 bit word format, SID can dump it with the DW option. DW groups memory together in 16 bit words, and following the INTEL format of storing word data, reverses the bytes. Supposing 200H contained:

```
DW   1234H
DW   BUFFER
```

If you use DDT or SID to dump 200H, you see:

```
0200 34 12 9C 01 ...
```

However if you DW200 with SID:

```
0200  1234 019C ....
```

NOTE the ASCII part of the dump is NOT turned around. An interesting side effect of the DW command, is that the display is less than 64 characters wide, so people with a VDM or other 64 wide display won't have the wrap-around problem they do with the regular D command of DDT or SID. Many thanks to Tom Cochran, who figured out how to patch SID to suppress the spaces between the bytes, so even the D command fits in 64 characters.

If you want to patch your version of SID to dump in 64 wide format, see if you have a JNZ 893 at address AA4, and if so, change it to JNZ 896. This will skip outputting the space.

(F)ill

The (F)ill memory command is just like DDT, except that the enhanced addressing and data formats supported by SID expand it significantly. For example, to fill a 100 byte buffer with spaces:

```
F.BUFFER,+#100,' '
```

says to start at BUFFER, through BUFFER + (decimal) 100, with spaces.

(An aside: there is a minor bug in DDT and SID when using the "F" command: if you specify the ending address as FFFF, they don't recognize this, and go "blasting" off back up past 0, blowing CP/M away. My version of SID is several years old, so you might check if later versions still have this bug.)

(G)o

The (G)o (execute) command is also just like DDT; i.e. it takes 0, 1, 2, or 3 operands to specify an optional starting address and 1 or 2 optional breakpoints. Passpoints (See P command) are shown as they are encountered, and control stops if the pass count is 0. Preceding the G command with a minus sign says to NOT display the passpoint data.

(H)ex

The (H)ex command is new with SID, and is very handy. Typing "H" dumps the symbol table, and each symbol's hex value. Typing "H" followed by a hex value (or a symbol) results in SID typing the value in hex, then in decimal. This serves the dual purpose of being a symbol table lookup (you type H.SYMBOL and SID replies with its hex value), and a hex-to-decimal conversion.

If you type "H" followed by two values, SID calculates and prints their sum and difference. The latter is most useful when loading a HEX or COM file with an offset. For example in using SYSGEN, my BIOS starts at AA00 and must be loaded into memory at 1E80. To figure out at what offset I have to read in the HEX file of my BIOS, I have to subtract AA00 from 1E80 to get 7480. This is the second value shown by executing:

```
H1E80,AA00
```

Just to finish off the example, I then:

```
ICBIOS.HEX
R7480
```

(I)nsert

The (I)nsert command is much more powerful than under DDT. DDT only fills in the first FCB, and does not place anything in the buffer at 80H. SID completely handles filling both names: one at 5CH, the other at 6CH; it also fully formats the raw command buffer at 80H. (If this makes sense to you, skip the remainder of the "I" description.)

Whenever you type a CP/M command, several things are set up in low memory before your COM file is loaded. First, all data in the command line is placed in a buffer at 80H. 80H itself contains the length of the command line followed by the data as you keyed in your command, minus the COM file name. Thus if you typed:

```
test alpha
```

80H would contain an 06, followed by " ALPHA", starting at 81H. For convenience, a 00H is placed the last byte keyed in. Note that the data you keyed is converted to upper case.

CP/M takes the names you typed after the command name, and places them in the system file control block (FCB) at location 5C. If a second name was given, it is placed at location 6C.

(L)ist

The (L)ist (or disassemble) command is like DDT. However, what is shown differs, in that if you have a symbol table, all word operands are looked up in the symbol table (see earlier example). Also, when an address at which the list is occurring is a symbol, that symbol is typed as a label (e.g. TEST:). In addition, such things as "CMP M" result in SID showing the value of memory, by placing "M=4D" (for example) next to the instruction. Very nice for "viewing" data which would otherwise be "invisible" while tracing.

If any of the formats of the L command are preceded by a mminus sign, the symbol table lookup and substitution is suppressed.

(M)ove

The (M)ove memory is exactly the same as DDT. Of course, the more powerful addressing modes and symbol table lookup of SID may be used in the operands.

(P)ass counter

The (P)ass counter command is totally unique to SID. A "Pass" is just what it sounds like: the program counter "passing" through a particular address. A "pass" is like an automatic breakpoint. It resets itself after being executed. Optionally, a "pass count" may be set, which says to literally, "pass" the location by, stopping only after the specified number of passes have occurred. As each passpoint is passed, a register and instruction dump (as shown by the T command) is shown. It may be suppressed by preceding the G command with a minus sign.

Let me give you an example: suppose you want to stop at the 15th time label TEST is executed. Just type:

```
p.test,#15
```

NOTE the use of the "#" to signify that 15 was to be taken as a decimal number. Then just type G, and every time TEST is executed, a full register dump will be shown, preceded by the current value of the "pass counter". When it counts to 0, a full breakpoint will occur. The passpoint remains set, but may be cleared with a "-P.TEST". Typing just "P" will show all current passpoints, and their counters. Typing "-P" will clear all passpoints.

(R)ead

The (R)ead program command reads in whatever file is named in the system FCB at 5CH. It need not be a .COM file. Like DDT, if an operand is specified, it implies an OFFSET to be added to the program as it is read in.

Additionally, a .SYM file may be read. This is done by placing it as the second operand on the I command, e.g.

```
ITEST.COM TEST.SYM
```

(S)et

Like DDT, the (S)et command allows patching bytes. In addition, words may be set into memory. Snnnn starts the byte set operation, SWnnnn starts in word set mode. In either case, the previous data item at the particular location is first shown. Pressing return leaves the old value. Pressing a "." terminates set mode.

(T)race

The (T)race command traces instructions one at a time. A special format which has two operands is used to call one of the SID utilities; for instance you might want to profile frequency of instruction execution by memory address, to help pick out what parts of your program are most frequently executed, so you may concentrate speed improvement coding to those areas.

(What? Ya, that's what I say. I either code my programs efficiently if they need to be efficient; I code them as easily as possible, if they need not be so efficient. I have not needed to locate my program's "hot spots" so far.)

A more significant utility is the one which permits backtracing. When the trace utility is loaded, it dynamically traps the last 256 trace lines, thus allowing you to run (must be in T or U mode) until a breakpoint is encountered; then you are able to see the last 256 steps. Now, THAT sounds handy. For example, if you have a program which has an error routine, and you can't imagine why it's being branched to, just place a breakpoint at the error routine, install the trace utility, and you should be able to solve your problem. The only possible difficulty I see is the time it might take to trace sufficient steps to encounter the error. Perhaps use of pass points to narrow down how many times a particular routine was executed, might help.

Preceding any of the T commands with a minus sign, says to suppress the symbol table lookup normally associated with tracing.

(U)ntrace

The (U)ntrace command functions just like DDT's, and is like the SID T command in that a utility may optionally be called at every instruction.

E(X)amine

The e(X)amine/set command allows you to examine and/or set A, BC, DE, HL, the stack pointer, the

program counter, or any of the individual bits of the PSW: (C)arry, (Z)ero, (M)minus, (E)ven parity, and (I)nterdigit carry. In case you haven't heard of that last bit, it is the one used by the DAA (Decimal Adjust Accumulator) instruction.

Again, my "review" has turned into more of a "tutorial", but I feel that by writing this way, I may be letting YOU evaluate SID. You can read my lines, and sometimes in between them, that I rate it tops. I repeat my recommendation: any serious assembler programmer should have MAC and SID in their "toolkit".

There are other debuggers available. For example, the Program of the Month club, sells a debugger called "DDS", for "Dynamic Debugging System". It supports significant features beyond SID; for instance, it is able to breakpoint when a particular memory location is referenced, or when the stack goes beyond certain bounds, etc. Very powerful. I bought an early version of DDS from Computer Mart of New Jersey, and was very impressed by it, but went strictly to SID because of SID's use of symbols. I do not know if the new version supports symbols.

The next edition of this review will feature Microsoft's MACRO-80 assembler and LINK-80, its link editor. As usual, I solicit your comments and/or criticisms about this review. I have received several letters asking me to review additional products, such as DDS II, and RMAC. However, my "style" (if it can be called that) is to review things which I am familiar with through my day-to-day usage. I certainly encourage users of these, and other products related to assembly language programming and debugging, to write their OWN reviews for inclusion in Lifelines. If you don't feel quite up to it, just write down your thoughts in any form, as detailed as possible, or even "talk" them into a cassette, and send them to me, in care of Lifelines. Thanks!

A Patch For muMATH Version 2.02

This patch provides the update of muMATH Version 2.02 to Version 2.03. in FUNCTION INT1 the next to last line was changed; replace PROPERTY INT, LOG.

```
FUNCTION INT1(EX1,
     % Local: % EX2, EX3, EX4, EX5),
     EX2: 1,
     WHEN EX3:INT2(), EX2*EX3  EXIT,
     TRGEXPD: LOGEXPD: NUMNUM: DENDEN: DENNUM: 30,
     PWREXPD: 6,
     EX1: EVAL(EX1),
     WHEN EX3:INT2(), EX2*EX3  EXIT,
     NUMNUM: DENDEN: DENNUM: -30,
     TRGEXPD: 7,
     EX1: EVAL(EX1),
     WHEN EX3:INT2(), TRGEXPD:-7, EX2*EVAL(EX3)  EXIT,
     EX2 * LIST ('INT, EX1, INDET),                          %**
ENDFUN $


PROPERTY INT, LOG, FUNCTION (EX1, EX2),                      %**
     WHEN EX2 EQ #E,                                         %**
          WHEN FREE (EX2:DIF(EX1,INDET), INDET),             %**
               EX1 * (LN(EX1)-1) / EX2 EXIT EXIT,            %**
ENDFUN $                                                     %**

STOP $
RDS() $
```

11/26/80

OOOOOOOOOOPS!

On Page 16 of the December issue of Lifelines in the first column mention is made of updating CP/M Version 1.4 to Version 2.20. The price of this update is $125, not $100 as stated in the article.

Catalogue CPMUG Volume 44

DESCRIPTION: CBASIC2 PROGRAMS

NUMBER	SIZE	NAME	COMMENTS
		CATALOG.44	CONTENTS OF CP/M VOL. 44
44.1	3K	ABSTRACT.044	Comments on contents of volume.
44.2	2K	ACCTFILE.BAS	Part of Osborne General Ledger.
44.3	5K	ANNTOT1.BAS	Part of Cunningham Ledger.
44.4	3K	ANNTOT1.INT	INT of above file.
44.5	1K	BINSERCH.BAS	Part of Osborne General Ledger.
44.6	12K	BUDGET1.BAS	Part of Cunningham Budget.
44.7	2K	BUDGET1.DOC	DOC of above file.
44.8	6K	BUDGET1.INT	INT of above file.
44.9	5K	BUDGETCH.BAS	Part of Cunningham Budget.
44.10	2K	BUDGETCH.INT	INT of above file.
44.11	1K	CBAS.SUB	SUBMIT file for compiling the Osborne programs.
44.12	10K	CRTFM.BAS	CRT File Maint. Utility for Osborne General Ledger.
44.13	4K	CRTFM.INT	INT of above file.
44.14	2K	CURSOR.BAS	Part of Osborne General Ledger.
44.15	2K	EXP1980.TOT	Cunningham Budget Data file.
44.16	1K	EXPENS80.JAN	Cunningham Budget Data file.
44.17	1K	FILEINIT.BAS	Utility for Osborne General Ledger.
44.18	1K	FILEINIT.INT	INT of above file.
44.19	2K	G/L.DOC	DOC on Osborne General Ledger.
44.20	2K	G/L-INFO.BAS	Part of Osborne General Ledger.
44.21	1K	G/L000.BAS	"
44.22	2K	G/L000.INT	"
44.23	6K	G/L010.BAS	"
44.24	4K	G/L010.INT	"
44.25	6K	G/L020.BAS	"
44.26	2K	G/L020.INT	"
44.27	6K	G/L030.BAS	"
44.28	5K	G/L030.INT	"
44.29	9K	G/L040.BAS	"
44.30	6K	G/L040.INT	"
44.31	12K	G/L050.BAS	"
44.32	6K	G/L050.INT	"
44.33	5K	G/L070.BAS	"
44.34	4K	G/L070.INT	"
44.35	8K	G/L080.BAS	"
44.36	5K	G/L080.INT	"
44.37	1K	GENINFO.BAS	"
44.38	7K	LEDGER1.BAS	Cunningham Ledger.
44.39	2K	LEDGER1.DOC	DOC of above file.
44.40	3K	LEDGER1.INT	INT of above file.
44.41	2K	POSTFILE.BAS	Part of Osborne General Ledger.
44.42	3K	ROBO.BAS	Cunningham Math Game.
44.43	1K	ROBO.DOC	DOC of above file.
44.44	2K	ROBO.INT	INT of above file.
44.45	5K	SUBS1.BAS	Part of Osborne General Ledger.
44.46	0K	TCGWOZ.TXT	Collected Great Works of Zoso *

* one anonymous reviewer's satirical opinion.

Catalogue CPMUG Volume 45 Abstracts

DESCRIPTION: Osborne/McGraw-Hill Payroll with Cost Accounting

NUMBER	SIZE	NAME	COMMENTS
45.1	1K	-CATALOG.045	Contents
45.2	2K	ABSTRACT.045	Abstract
45.3	1K	P/R.SUB	SUBMIT file for CBAS2 compilation.
45.4	2K	P/R000.BAS	Menu Program for P/R.
45.5	5K	P/R010.BAS	Part of P/R.
45.6	13K	P/R020.BAS	Part of P/R.
45.7	3K	P/R030.BAS	Part of P/R.
45.8	2K	P/R040.BAS	Part of P/R.
45.9	10K	P/R050.BAS	Part of P/R.
45.10	4K	P/R06A.BAS	Part of P/R.
45.11	8K	P/R06B.BAS	Part of P/R.
45.12	4K	P/R070.BAS	Part of P/R.
45.13	2K	P/R080.BAS	Part of P/R.
45.14	8K	P/R090.BAS	Part of P/R.
45.15	9K	P/R100.BAS	Part of P/R.
45.16	8K	P/R110.BAS	Part of P/R.
45.17	3K	P/R120.BAS	Part of P/R.
45.18	4K	P/R130.BAS	Part of P/R.
45.19	6K	P/R140.BAS	Part of P/R.
45.20	3K	P/R150.BAS	Part of P/R.
45.21	2K	P/R160.BAS	Part of P/R.
45.22	3K	P/R170.BAS	Part of P/R.
45.23	8K	P/R180.BAS	Part of P/R.
45.24	4K	P/R190.BAS	Part of P/R.
45.25	7K	P/R200.BAS	Part of P/R.
45.26	9K	P/R210.BAS	Part of P/R.
45.27	6K	P/R220.BAS	Part of P/R.
45.28	5K	P/R230.BAS	Part of P/R.
45.29	9K	P/R240.BAS	Part of P/R.
45.30	3K	P/R250.BAS	Part of P/R.
45.31	3K	P/R260.BAS	Part of P/R.
45.32	6K	P/R270.BAS	Part of P/R.
45.33	15K	P/R280.BAS	Part of P/R.
45.34	4K	P/R290.BAS	Part of P/R.
45.35	10K	P/R291.BAS	Part of P/R.
45.36	11K	P/R300.BAS	Part of P/R.
45.37	5K	P/R310.BAS	Part of P/R.
45.38	3K	P/R320.BAS	Part of P/R.
45.39	7K	P/R321.BAS	Part of P/R.
45.40	6K	P/R330.BAS	Part of P/R.
45.41	3K	P/R340.BAS	Part of P/R.

CPMUG Volume 44

PROGRAMS:

BUDGET1--Enters budget information to file
LEDGER1--Totals month file and reports
ANNTOT1--Reports the year to date
BUDGETCH--Change file information (errors)

AUTHOR:
Pat Cunningham, San Antonio, TX
SOFTWARE: CBASIC
HARDWARE: 96 column or greater printer, if you wish hard copy.
DOC FILES:
LEDGER1.DOC & BUDGET1.DOC

The program is nicely written with good commenting. All of the programs run and seem to work properly. Error trapping could be improved. LEDGER1 allows a rerun at the end, but this bombed for me. The main weakness of the program is its inadequate user prompts. The BUDGETCH expects the user to remember the 20 letters that reference the expense categories. The category descriptions are available as a user call from the BUDGET1 program. The program would be improved if it provided the information without request in both BUDGET1 and BUDGETCH. All in all this is a usable program to enter monthly expenses for the home--after you get used to the required input formats. LEDGER1 sign-off is a reference to a mythical state--somewhat akin to Camelot combined with Devil's Island.
Ken Prevo

PROGRAM: ROBO
AUTHOR:
Pat Cunningham, San Antonio, TX
SOFTWARE: CBASIC
HARDWARE: No special requirements
DOC FILE: ROBO.DOC

ROBO is a math drill (addition) for preschoolers and first-graders. It provides a line of num-

bers that I guess saves on the fingers and toes method I use. This might make some teachers uphappy, but is a nice little helper for those learning beginning math.
Ken Prevo

AUTHOR: Anon
PROGRAM: The Collected Great Works of Zoso
FILE LENGTH: OK

It is rumored the sequel will be THE INANE REMARKS OF ZOSO THE GREAT. A reviewer is currently being solicited. Anyone with quad-density interested?
POGO

PROGRAMS: GENERAL LEDGER

This is the General Ledger series of programs published by Osborne/McGraw-Hill.

You WILL need to purchase the book GENERAL LEDGER CBASIC by Lon Poole, with Mary Borchers, Martin McNiff, and Robert Thomson. It is published by Osborne/McGraw-Hill and is available in many computer stores and some book stores. If you are unable to find it any other way, you can usually order it through a bookstore.

This disk contains a file called CBAS.SUB which is a submit file for compiling the programs on this disk in proper sequence. Some of the BAS files you see on the disk are not listed in the .SUB file because they are subroutine type programs that are loaded with the main programs using the CBASIC2 $INCLUDE statement.

Two of the programs on the disk, CRTFM and FILEINIT are utilities that you will use the first time you set up your data files, and used seldom after that.

You will want to modify the CURSOR.BAS program for your CRT -- CURSOR.BAS is set up for a Hazeltine terminal and if you don't have a Hazeltine, then you'll have to make some changes.

SUBS1.BAS also has some cursor positioning routines in it that you may need to modify. These modifications will require a good deal of familiarity with the manual and the operation of your CRT as opposed to a Hazeltine. Familiarity with CBASIC2 programming is also required.

After you have compiled the programs, you will want to put all the .INT files on a blank disk, perhaps with CRUN2.

Refer to the book for operating instructions. G/L000 is the menu program.

Good luck,
Jim Mills

(Some of these comments are taken not from the abstract, but from the file G/L.DOC. Lifeboat Associates carries the books referred to above; they are priced at $20 each.-Ed.)

Abstracts

CPMUG Volume 45

Osborne/McGraw-Hill Payroll with Cost Accounting

It is assumed that the user of this disk will also have obtained CP/M User Group volumes 43 and 44, Osborne's Accounts-Payable/Accounts-Receivable and General Ledger programs. This is important in this case as we ran out of room on the disk (Oh, for a standard for double density!) and some of the subroutine.BAS files are on the other disks. I refer specifically to the following:

CURSOR.BAS
SUBS1.BAS
GENINFO.BAS

In addition it seems that our source (he who typed-in all these programs) missed some of the P/R subroutines which may be found in the book PAYROLL WITH COST ACCOUNTING published by Osborne/McGraw-Hill:

PRNMASK.BAS
FIND-EOF.BAS
MSTRIN.BAS
MSTROUT.BAS
HISTORY.BAS
P-RSEARC.BAS
JOBFILE.BAS

Also, one of the "main" programs is missing from this disk:

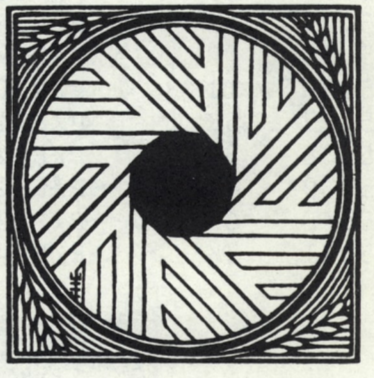
MSTRINIT.BAS

The above files are not overly lengthy and may be keyed in without too much difficulty. It is assumed that the user will compile these programs and place the leaving the .BAS files elsewhere as they are not needed for actual operation.

There is a file P/R.SUB on this disk for ease of compiling. It is assumed that the person installing these programs for usage will have some familiarity with CBASIC2 conventions.

The CP/M Users' Group assumes no liability or responsibility for these programs or the results of using these programs.
James K. Mills

Coming soon in Lifelines will be a series of articles on the packages featured in Volumes 43, 44, and 45. Written by experts from Osborne/McGraw-Hill, these reviews will describe the packages in detail and provide the user with valuable guidelines as to their use.



New Products

FILETRAN by
Business Micro Products

FILETRAN is a complete set of disk utilities and file translation routines for owners of TRS-80 Model I microcomputers with CP/M and TRS-DOS (including all compatible variations). The transfer of complete files between TRS-DOS and CP/M disks optionally in either direction is accomplished via a simple interactive process. Additional features are included which allow the user to display selected sectors of a disk containing either CP/M or TRS-DOS data or examine memory during a transfer. A utility which will scan ASCII Level II BASIC files and display lines known to cause problems in Microsoft BASIC 5.0 is also provided. This a great time saver when translating BASIC programs between operating systems. Finally, a FILES command and a DIR command are included to display the CP/M or TRS-DOS directories respectively.

FILETRAN is delivered configured for the TRS-80 Model I terminal. A configuration program is provided for those wishing to use a different terminal. FILETRAN assumes that drives A and B are standard mini-drives (35 tracks, 10 or 18 sectors, 256 or 128 bytes per sector) and C and D are standard 8 inch drives (77 track, 26 sectors, 128 bytes per sector). This does not preclude the use of another disk configuration since these assumptions are only used to provide warning messages.

A version is also available for TRS-80 model II owners which includes all of the features mentioned above with the exception of CP/M to TRS-DOS transfers.

Because of the specific disk drivers included, there is no guarantee that FILETRAN will perform correctly on TRS-80 model I's to which hardware modifications have been made.

The versions available from Lifeboat Associates and their prices follow: 1.2 for Model I with Lifeboat CP/M, to CP/M only--$99; 1.4 for Model I with Lifeboat CP/M, bi-directional--$149; 1.5 for Model II with Lifeboat CP/M, to CP/M only, $99.

Microspell
by Bob Lucas

This program is designed to virtually eliminate spelling errors and typos from your texts. Equipped with a dictionary of 20,000 words, it scans your documents for dubious spellings. The user may expand the Microspell dictionary, dependent upon the space available. Up to about 13,000 words in a 64K system and 3,000 words in a 48K system can be added to the user's lexicon.

Microspell analyzes a text, stopping at unfamiliar words. As the document is scanned, Microspell displays plural expansions, suffix expansions, and the contexts of questionable words; these display features may be switched off, if so desired. Obviously, this tool will not recognize correctly spelled words which are inadvertently substituted for others (i.e., the use of "it" for "in").

When Microspell halts at a word, the user is offered the following options:

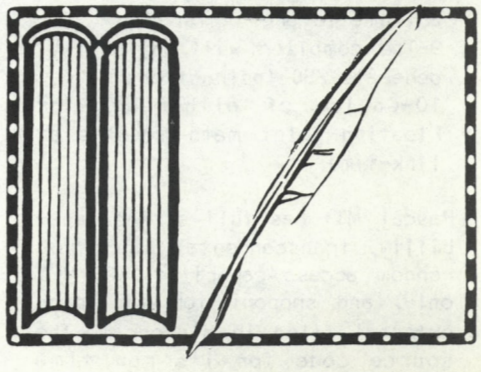
1-The user may (A)ccept the target word.
2-The user may (I)nsert the target word into the dictionary. In this fashion, the user can make sure that any jargon or special expressions and acronyms required are later recognized by the spelling tool.
3-The targeted word may be (R)eplaced with a word typed in on the console by the user.
4-The user may ask that the context be (D)isplayed.
5-The (P)robe command allows the user to investigate the words contained in Microspell corresponding to a questionable one. This is especially valuable to a user who knows the approximate spelling of a word. Microspell will display like-spelled words from which the user may choose, thereby eliminating the use of a dictionary.

One of Microspell's features is to provide a series of choices in a menu-like fashion when given an improperly spelled word. The user may in response employ one of the commands above, specify the correct words from the selections offered, generate more guesses, or probe the dictionary for still more possibilities.

Microspell provides an EXCeption file, which may be optionally suppressed. This file stores the correct forms of mispelled words for each document scanned. Words recognized by suffix stripping can also be sent to this file, which can be used as an auxiliary lexicon.

Microspell can be asked to accept as correct words of all upper case letters, obviating the necessity for adding large numbers of acronyms to its vocabulary.

Microspell is available from Lifeboat Associates at a price of $249, $20 for the users manual.



(continued overleaf)

Pascal MT+ Compiler
by MT Microsystems

The Pascal MT+ Compiler supports the ISO Standard Pascal. It permits the user to separately compile Pascal and assembly language modules which can be combined to produce a final program. The run time overhead ranges from 256 to 1200 bytes.

Pascal MT+ is designed to be faster than interpreted and other native code Pascals. It incorporates optimization steps taken during compilation to remove redundant PUSH/POP sequences and use single increment and decrement instructions when adding or subtracting small literal numbers. The disassembler interleaves the Pascal source code and symbolic assembly code for greater programming efficiency.

In addition, Pascal MT+ includes the following features:

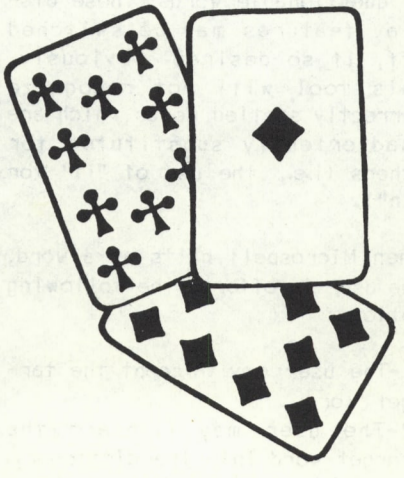
1-Predeclared arrays INP and OUT directly access I/O ports.
2-An INLINE facility is provided for inserting assembly language between the Pascal statements and can be used to generate constant tables.
3-ABSOLUTE assembly language procedure declarations may be employed for pre-assembled routines.
4-INTERRUPT procedures are included.
5-Chaining permits the transfer of control from one program to another.
6-There is an ELSE clause on the case statement.
7-With redirected I/O facilities, user-written character level I/O drivers can be called via READ and WRITE statements.
8-HEX literal numbers are supported; business and scientific arithmetic are included.
9-The compiler will optionally generate Z80 instructions.
10-Choice of either BCD or Floating point math modules at link-time

Pascal MT+ has full string capability, transcendental functions, random access capability (CP/M 2 only) and support for including external files in a program. The source code for its run time library is included allowing straight forward user customization. Built-in procedures and functions are as follows: bit test, clear, set; return high or low byte; shift left and right; return the address of a data item or routine; return the size of a data item; enable, disable interrupts; standard file I/O, random read and write to files.

A symbolic debugger traces one or more lines of Pascal code, or executes the program up to a line number or symbolic breakpoint. The debugger displays the name of each procedure and function. The contents of simple and complex variable can also be displayed by name. The debugger is usable in a ROM environment.

Along with the Pascal compiler and interactive symbolic debugger, the package includes a Librarian program to manage module libraries, sample programs, the Pascal library and utility routines, and linker.

Pascal MT+ requires 52K of memory. It is priced at $250, available from Lifeboat Associates. Pascal/MT will continue to be offered by Lifeboat for users with insufficient memory (32k) to run Pascal MT+. Current owners of Pascal/MT can update to Pascal MT+ for the unadvertised special rate provisionally set at $75.



PL/I-80 by
Digital Research

PL/I-80 is a complete software package for application programming under CP/M and MP/M. It is based upon the new subset G language defined by the ANS PL/I Standardization Committee X3J1. It is a tool for the effective, speedy development of large scale structured applications in the microcomputer environment. Included in the package are the following:

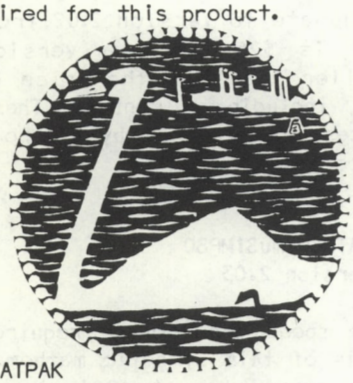
1-The PL/I-80 compiler
2-The LINK80 linking loader
3-RMAC, a macro assembler capable of generating relocatable modules
4-LIB-80, a comprehensive library manager
5-XREF, a cross reference listing generator
6-A large number of sample programs and other useful modules
7-A Language Manual, Application Guide, and Operator Guide.

The package supports the Microsoft REL format for relocatable modules as well as its own IRL (indexed relocatable library) format providing both compatibility with other compilers (FORTRAN, BASIC, COBOL, Pascal) and fast library searches when operating in it own environment. The Loader provides for the effective use of overlays, a first for Microcomputer compilers, allowing very large applications to be developed. Furthermore the linker will produce PRL (Page relocatable) files, a must for anyone working with MP/M.

In addition to the standard PL/I hardware independent features, DR has provided several assembly language modules which allow a user to take advantage of the specific features of CP/M. Among these is a package that will allow CP/M system calls to be initiated from an application program. The Library manager contains commands to replace or delete modules in a single pass as well as the standard functions included in other similar products. Furthermore it will operate on any REL format library as well as DR's IRL format and will perform translation between

these two formats.

PL/1 is available from Lifeboat Associates at a special price of $500; this is a limited offer, effective only until February 28th, when the price will rise. A prior license agreement is required for this product.



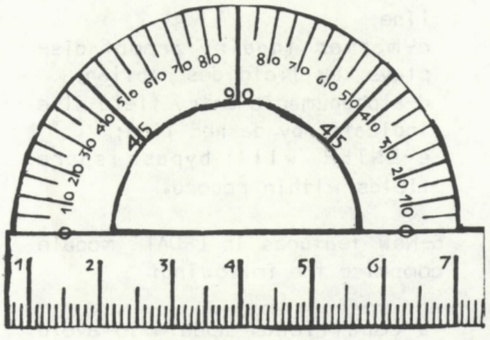
STATPAK
by Northwest Analytical

STATPAK is a statistics software library designed for microcomputers using Microsoft BASIC version 4.51 and CP/M. It provides the user with a full scale statistics package similar to those found on time-shared and mainframe systems.

The package is divided into two major sections; one dealing with the entry and manipulation of data files, including editing, merging, and selection of data subsets; the other providing all of the major statistical functions used in normal practice.

Functions include: combinatorial analysis; Bayes Formula; repeated Birthday Problem; Standard Deviation; Standard Error; Arithmetic, Geometric, Harmonic and Quadratic means; first, second, third, and fourth moments about mean; Coefficients of Skewness, Kurtosis and Variation; Moving Averages; Standardized Scores; Generalized Mean; Histogram. Discrete distribution functions include: Binomial; Poisson; Negative Binomial; Hypergeometric. Continuous distribution functions include: Khrgian and Mazin; Exponential; Normal; Inverse Normal; Chi-squared; F; T; Bivariate Normal; Logarithmic Normal; Gamma; Incomplete Gamma. Also contained are programs for linear, exponential, logarithmic, power, multiple linear and polynomial regression analysis. Mean testing by both T statistics and one sample tests are provided. Tests are also given for the analysis of survey data and contingency tables. Several Non-Parametric tests are also included as well as analysis of variance for up to 30 levels of replications.

Statpak is obtainable from Lifeboat Associates for $495; the documentation is priced at $30 when purchased alone.



Ultrasort-II
by Computer Control Systems

Ultrasort is a sort utility, written in 8080 assembly language, that can be used in either a "stand-alone" configuration or can be called as a "overlay" subroutine by a high level language such as CBASIC. Other sort packages show a substantial degradation of sort speed when used as subroutines because the coresident application program limits the amount of buffer space that can be allocated. Fast sort times are maintained by Ultrasort in this mode of operation since it writes the TPA (user program) to disk prior to loading itself, thereby greatly expanding internal buffer space.

Ultrasort contains many special features such as:

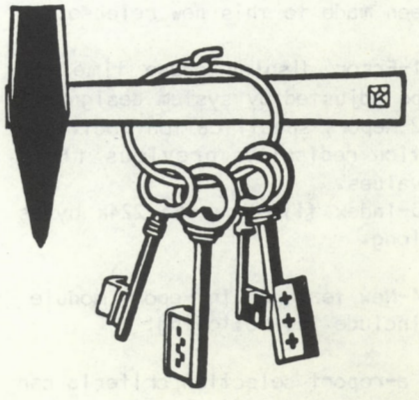
1-Sort/Merge/Select capability
2-Finds number of logical records in a file
3-Dynamically creates sort parameter files
4-Has virtually unlimited record size (tested with records in excess of 5000 bytes)
5-Sorts on 5 keys, independently ascending or descending
6-Variable or fixed length fields
7-Supports Floating point numbers
8-Skip first record option
9-DELETES/RETAINS records comparing up to four independent select keys based on select key being less than, equal to, or greater than the test field
10-Optional select key "OR" function
11-Prompted disk change capability

Ultrasort is priced at $195 and $25 for the documentation alone, available from Lifeboat Associates. V/SORT will be discontinued by Lifeboat. Current owners of V/SORT can purchase Ultrasort for the reduced rate of $ 75 ; this offer will be good for a limited time only.

Unlock

Unlock is a utility provided for owners of Microsoft BASIC release 5.2. It decodes programs that have been saved in protected "P" format back into their original unencrypted form. This can be a lifesaver if the original has been lost.

Lifeboat Associates sells Unlock for $195; the documentation alone costs $25.



New Versions

BUG/uBUG
Version 2.02

The following bugs have been fixed:

1-The BUG "I" command has been fixed so as to not destroy the user's program counter, <PC, when executing a non-transfer type instruction. In previous versions transfer-of-control instructions (i.e. JMP, CALL, RET, etc.) updated the <PC properly, but non-transfer instructions left garbage.
2-The BUG/uBUG "O" command has been fixed to handle lower case filenames properly. In previous versions, the lower case extensions "hex" and "rel" were not recognized, and the file was loaded as if it were of type COM.
3-BUG/uBUG has been updated to save/modify/restore locations [8, 9, 10] as well as [5, 6, 7] if the current operating system is PDOS, but leave [8, 9, 10] alone otherwise. Previous versions always modified both sets of locations, causing problems in operating environments which use RST 1 for other than system calls.

The price of the Bug/uBug update is $25, from Lifeboat Associates.

CBS
Version 1.2

The following enhancements have been made to this new release:

1-Error display pause time can be adjusted by system designer.
2-Report specification modification redisplays previous field values.
3-Index files can be 224k bytes long.

4-New features in Report module include the following:

a-report selection criteria can be modified at run time;
b-calculation operation to add range of fields/columns;
c-subtotalling on three levels;
d-running total report columns;
e-report to disk after destination requested from operator.

5-New features in ENTER module include:

a-ability to duplicate data entered in previous record;
b-CTL-X to delete entire input line;
c-master inquiry prompt displays key field description;
d-alphanumeric entry field size indicated by dashed line;
e-ENTER will bypass spare fields within record.

6-New features in UPDATE module comprise the following:

a-conditional update to avoid replacement if field empty;
b-all-add option to force new record creation for all transactions;
c-mass delete option for all records whose key is located;

7-New features in CREIDX module:

a-partial keys are permitted;
b-CREIDX will permit duplicates in all index files, allowing a master file damaged by an aborted CLEANUP to have its primary index recreated through a CREIDX step.

8-CLEAR module to recover and return unused space to O/S.
9-Password protection is included for system/menu item access.
10-Date, batch and user number are included in all data records.
11-A new improved users manual is now available.

The price of an update to this version, available from Lifeboat Associates, is $100. This includes the new manual.

muLISP-80
Version 2.03

The information which arrived with this new release read "bug fixes". Since there is no way to imply which bugs have been fixed, I cannot recommend that anyone purchase this update. If you are experiencing problems and don't mind risking your money it may be worth a try. In the future I would hope that authors will include information along with updates that is of benefit to users.

To update to Version 2.02 from 2.00 is $25. From a version earlier than 2.00 the price is $65, including a manual. These prices are provided by Lifeboat Associates.

muMATH80/muSIMP80
Version 2.03

The speed and storage requirements of this symbolic mathematics system have been improved. All system modules can now reside together in a 60k machine. Furthermore, system construction time has been reduced by several orders of magnitude.

These new modules are included:

1-SIGMA.ALG provides facilities for determining closed form sums and products using classic capital sigma and capital pi notation. Either numeric, symbolic or indefinite limits may be specified.
2-TAYLOR.DIF is a function that uses repeated differentiation and substitution to yield a truncated Taylor series expansion of an expression with respect to one of the unbound variables therein.
3-LIM.DIF provides facilities for finding the one-sided limit of a mathematical expression as one of its variables approaches a value. Besides ordinary numeric or nonnumeric values, the boundary point can be designated as plus infinity or minus infinity.

To update your Version 2.02 of muMATH to 2.03, see the patch on page 5 of this issue. To update to Version 2.02, the price from Lifeboat Associates is $70, including a new manual.

Pascal/Z Version 3.21

The following changes have been made in this new release:

1-One letter file names are now accepted.
2-The user is reprompted after a 'String too Long' runtime error.
3-The ROUND and TRUNC functions return correct values for numbers between -32767 and -16484.
4-READLN with no parameters functions properly.
5-The LN function returns an error message when it is called with a negative argument.
6-Several bugs involving the use of Direct Access files have been corrected.
7-It is no longer necessary to append a <CR> to a variable length string when using it as a file name for RESET or REWRITE.
8-The type checking of STRING and CHARACTER data has been improved.
9-The use of structured functions as parameters passed to other functions is now correctly implemented.
10-The LST file generated by the compiler can now be directed to either the console or the printer.
11-Pointers which point to fields within records without using the WITH statement now function correctly.
12-Boolean expressions comparing two Boolean constants now evaluate correctly.
13-A bug which occurred when Boolean expressions were contained within FOR or REPEAT statements has been fixed.
14-Forward declared functions now work.
15-Appropriate error messages are returned if FILE variables are used in assignment statements, as FIELDs of RECORDs, or as value parameters.
16-Null STRING and null filename now cause runtime error messages.
17-Error recovery has been improved.
18-EOF or EOLN can now be correctly assigned to a Boolean field of a RECORD when passed by reference using a WITH statemeant.
19-The compiler now checks for a quote at the end of a STRING.
20-The Title of the program being compiled is now output to the console when the main program compilation begins.
21-The handling of bytes in CASE statements has been improved.
22-A bug involving a pointer field of a global RECORD which points to a field of a RECORD has been fixed.
23-The STRING constant comparison algorithm has been fixed.
24-The range checking of SETs has been improved.
25-The subranges for SETs are now explicitly limited to within the range 0..255
26-RENAME and ERASE from within Pascal programs has been added.
27-Correct range checking of SETs of characters, enumeration types, and Booleans is now implemented.
28-Eight character filename followed by four character extensions are now correctly handled.
29- When using the radian, sine and cosine to find a tangent, Pascal/Z, equated the sine and cosine so that correct results could not be obtained. This bug has been corrected in Version 3.21.

Lifeboat Associates charges $60 for an update of a version prior to 3.0. An update from 3.2 is priced at $25; if you have version 3.0, an update to 3.21 will cost $40, including addenda to the manual.

PLINK Version 3.24

The following bugs have been fixed in this new release:

1-Microsoft COBOL modules could not be linked if more than one COBOL module was included in the program. A new diagnostic error, #52, will occur if a COBOL problem is encountered.
2-Digital Research PL/1 modules sometimes caused Plink to go into an infinite loop. They are now handled correctly.
3-The /IDENT option did not work; it gave error #2.
4-If a REL file was to be output, Plink would insist on having a defined starting address. It now creates a module with no starting address if none is defined.
5-If two Plink commands were put into the same file to be used via an "@", Plink would die. Now, a semicolon ending a command may appear in an "@" file.
6-Microsoft assembler modules would cause error #46 if the programmer tried to set the location counter below address 100H (for instance setting it to zero to define field offsets within a record). Now, the error won't occur unless an attempt is made to actually load code or data below address 100H.
7-The symbol .MAIN. was not used as the program starting address if a module had one; now it is.
8-Command input from LNK files is no longer echoed to the console as it is read in.

Plink can now handle all of the current releases of the Microsoft and Cromemco compilers. Digital Research PL/1 files are supported but the indexed library files (IRL) are not (these can be converted to REL files using DR's LIB program). Furthermore, PL/1-80 overlays are not supported. The MT Microsystems' Pascal MT+ compiler is supported only if normal files are created; the ERL files used by their debugger are not handled. Future versions of Plink may handle some of these non-standard formats.

A Plink update from Lifeboat Associates costs $25.

T/MAKER Version 1.4

The following improvements have been made with this version:

1-There are now automatic default tab settings for quick tabulation.
2-The cursor control logic has been changed so that wraparound does not occur when typing characters in the Editor.
3-The manual has been redone, including more examples and a tutorial.
4-A Net Present Value Function has been added to Compute. This function, which calculates the discounted value of money, is very valuable in any financial computation involving money over a period of time.

The T/MAKER update to the new version is priced at $50, from Lifeboat Associates.

Version List

12/15/80
Asterisks (*) indicate new products or new versions.

Product Name	Standard Version	Modified Version	Operating System	Processor	Memory Required	Price	Available From	Special Notes
A-NATURAL Assembler Package	1.1		CP/M	8080		330/15	LBA	
A3+ Development Package			CP/M	Z80		409/40	LBA	
A4 Development Package			CP/M	Z80		299/40	LBA	
Accounts Payable/Graham Dorian	1.11	1.11	CP/M	8080	48K	805/40	LBA	Needs CBASIC2
Accounts Payable/Structured Sys	1.3		CP/M	8080	52K	820/40	LBA	Uses IT WORKS run time pkg.
Accounts Payable/Peachtree	10-10-80		CP/M		48K	530/40	LBA	Needs Microsoft BASIC 4.51
Accounts Receivable/Graham Dorian	1.08	1.08	CP/M	8080	48K	805/40	LBA	Needs CBASIC2
Accounts Receivable/Peachtree	10-10-80		CP/M	8080	48K	530/40	LBA	Needs Microsoft BASIC 4.51
Accounts Receivable/Structured Sys	1.4C		CP/M	8080	56K	820/40	LBA	Uses IT WORKS run time pkg.
ALDS TRSDOS	3.38		TRSDOS	8080	32K	80/25	LBA	Is TRS-DOS Macro-80
ALGOL 60 Compiler	4.8C		CP/M	8080	32K	199/20	LBA	
ANALYST	1.0		CP/M	8080	52K	250/15	LBA	Needs CBASIC2
APL/V80 Compiler	3.1		CP/M	Z80	48K	500/30	LBA	Needs APL terminal
Apartment Management	1.03	1.03	CP/M	8080		805/40	LBA	Needs CBASIC2
ASM by XITAN	3.11		CP/M	Z80		69/20	LBA	
*BASIC-80 Compiler	5.2	5.2	CP/M	8080	40K	325/25	LBA	
BASIC-80 Compiler		5.1	TRSDOS		64K	400/25	LBA	TRS-80 Model II only
*BASIC-80 Interpreter	5.2	5.2	CP/M	8080	40K	325/25	LBA	Includes Versions 4.51 & 5.2
BASIC Utility Disk	2.0	2.0	CP/M	8080		50/35	LBA	
BSTAM Communication System	4.4	4.4	CP/M	8080		150/10	LBA	
*BDS C Compiler	1.42	1.42T	CP/M	8080	32K	145/25	LBA	Includes "C" book
Whitesmiths C Compiler	2.0		CP/M	8080	60K	630/30	LBA	
BSTMS	1.2	1.2	CP/M	8080	24K	200/15	LBA	
*BUG / uBUG Debuggers	2.02		CP/M	Z80		129/25	LBA	
Cash Register	2.0	2.0	CP/M	8080		805/40	LBA	
CBASIC Compiler	2.07	2.16	CP/M	8080	32K	120/15	LBA	2.16 is modified vers. 2.06
CIS COBOL Compiler	4.3,1		CP/M	8080	48K	850/50	LBA	
FORMS 2 COBOL Form Generator	1.1,6		CP/M	8080		200/20	LBA	
*CBS Applications Builder	1.21		CP/M	8080	48K	395/40	LBA	No support language needed
COBOL-80 Compiler	4.01	4.01	CP/M	8080	48K	700/25	LBA	
DATASTAR Information Manager	1.1		CP/M	8080		350/35	LBA	
Datebook	1.05		CP/M	8080	32K	295/25	LBA	Needs 80x24 terminal
DESPOOL Print Spooler	1.1A		CP/M	8080		80/10	LBA	
DISILOG Z80 Disassembler	4.0	4.0	CP/M	Z80		65/10	LBA	
DISTEL Z80/8080 Disassembler	4.0		CP/M	8080/Z80		65/10	LBA	
EDIT Text Editor	2.06		CP/M	Z80		129/25	LBA	
EDIT-80 Text Editor	2.0		CP/M	8080		89/15	LBA	
ESQ-1	1.0		CP/M	8080		1495/50	LBA	
*ESQ-1 DEMO	1.0		CP/M	8080		75/50	LBA	
FORTRAN-80 Compiler	3.36A	3.36A	CP/M	8080	36K	425/25	LBA	
FORTRAN PACKAGE	3.38		TRSDOS	8080		150/25	LBA	
Floating Point Processor	2.1	2.1	CP/M	8080	28K	250/10	LBA	In source,goes with RAID
FORTRAN TRS	3.38		TRSDOS	8080		80/25	LBA	
General Ledger by Graham Dorian	1.09	1.09	CP/M	8080	48K	805/40	LBA	Needs CBASIC2
General Ledger by Peachtree	10-10-80		CP/M	8080	48K	530/40	LBA	Needs Microsoft BASIC 4.51
General Ledger by Structured Sys	1.4C		CP/M	8080	52K	820/40	LBA	No longer needs CBASIC
GLECTOR Accounting System	2.0		CP/M	8080	52K	350/25	LBA	Use w/ CBASIC2 & Selector III
IBM/CPM	1.1		CP/M	8080		175/5	LBA	
INVENTORY by Graham Dorian	1.0	1.0	CP/M	8080		555/40	LBA	Needs CBASIC2
INVENTORY by Peachtree	10-10-80		CP/M	8080	48K	660/40	LBA	Needs Microsoft BASIC 4.51
*INVENTORY by Structured Sys	1.0C		CP/M	8080	52K	820/40	LBA	No longer needs CBASIC
JOB COSTING	2.02	2.02	CP/M	8080	48K	805/40	LBA	Needs CBASIC2
KBASIC Interpreter	2.03		CP/M	8080	48K	585/45	LBA	
KISS File Management System	2.03		CP/M	8080	40K	335/23	LBA	
LETTERIGHT Text Editor	1.0		CP/M	8080	52K	200/25	LBA	
LEVEL 3 BASIC by G2			TRSDOS	8080		45	LBA	Cassettes
LINKER by XITAN			CP/M	Z80		69	LBA	
MAC Macro Assembler	2.0		CP/M	8080		120/15	LBA	
MACRO-80 Macro Assembler Package	3.37	3.37	CP/M	8080		149/15	LBA	

LBA=Lifeboat Associates

Version List

12/15/80
Asterisks (*) indicate new products or new versions.

Product Name	Standard Version	Modified Version	Operating System	Processor	Memory Required	Price	Available From	Special Notes
Magic Wand	1.0		CP/M	8080	32K	395/40	LBA	
MAGSAM III	4.1		CP/M	8080	32K	145/25	LBA	
MAGSAM IV	1.0		CP/M	8080	32K	295/25	LBA	
MAILING ADDRESS Mail List System	8-13-80		CP/M	8080	48K	530/30	LBA	
Mail-Merge	2.1		CP/M	8080		150/25	LBA	
*Master Tax	1.0		CP/M	8080	48K	995/30	LBA	
*Microspell	1.0		CP/M	8080	48K	249/20	LBA	
MP/M Operating System	1.1		MP/M	8080	32K	300/50	LBA	
*Mu LISP-80 Compiler	2.03		CP/M	8080		200/15	LBA	
*Mu SIMP / Mu MATH Package	2.03		CP/M	8080		250/20	LBA	Is MuMATH 80
NAD Mail List System	3.0C		CP/M	8080	49K	100/20	LBA	
Nevada COBOL	1.403	1.403	CP/M	8080	32K	149/25	LBA	
PASM Assembler	1.02		CP/M	Z-80		129/25	LBA	
Pascal/M	3.2		CP/M	8080	56K	175/20	LBA	
PASCAL/MT Compiler	3.2		CP/M	8080	32K	250/30	LBA	
*PASCAL/Z Compiler	3.2-1		CP/M	8080	56K	395/25	LBA	
*Payroll by Peachtree	11-7-80		CP/M	8080	48K	530/40	LBA	Needs Microsoft BASIC 4.51
Payroll by Structured Sys	1.0C		CP/M	8080	60K	820/40	LBA	No longer needs CBASIC
PLINK Linking Loader	3.22		CP/M	Z-80		129/25	LBA	
POSTMASTER Mail List System	3.2	3.2	CP/M	8080	48K	150/20	LBA	
Property Manager	10-10-80		CP/M	8080		925/40	LBA	Needs Microsoft BASIC 4.51
QSORT Sort Program	1.5		CP/M	8080	48K	100/20	LBA	
RAID	4.7.3	4.7.3	CP/M	8080	28K	250/25	LBA	
RECLAIM Disk Verification Program	2.1		CP/M	8080		80/5	LBA	
SELECTOR-III-C2 Data Manager	3.21NC	3.21	CP/M	8080	48K	295/20	LBA	
SID Symbolic Debugger	1.4		CP/M	8080		105/15	LBA	
SMAL/80 Programming System	3.0		CP/M	8080		75/15	LBA	
STRING BIT FORTRAN Routines	1.02	1.02	CP/M	8080		65/15	LBA	
STRING/80 bit FORTRAN Routines	1.16		CP/M	8080		95/20	LBA	
STRING/80 bit Source	1.16		CP/M	8080		295/NA	LBA	
SUPER SORT I Sort Package	1.5		CP/M	8080		225/25	LBA	Max. record length=4096 bytes
SUPER SORT II Sort Package	1.5		CP/M	8080		175/25	LBA	Max. record length=4096 bytes
SUPER SORT III Sort Package	1.5		CP/M	8080		125/25	LBA	Max. record length=4096 bytes
T/Maker Data Calculator	1.3		CP/M	8080	48K	275/25	LBA	Requires CBASIC2
TEX Text Formatter	1.1		CP/M	8080	36K	105/15	LBA	
TEXTWRITER-III Text Formatter	3.5	3.5	CP/M	8080	32K	125/20	LBA	
TINY C Compiler	800102C		CP/M	8080		105/50	LBA	
VSORT Sort Program	1.8	1.8	CP/M	8080	48K	175/20	LBA	
WHATSIT? Data Manager	2.04		CP/M	8080	44K	175/25	LBA	
WORDMASTER Text Editor	1.07		CP/M	8080	40K	145/25	LBA	
WORDSTAR Word Processor	2.1		CP/M	8080	48K	445/40	LBA	
MAIL MERGE Printer Overlay	2.1		CP/M	8080	48K	575/40	LBA	
WORDSTAR Customization Notes	2.1		CP/M			95	LBA	
*XASM-18 Cross Assembler	1.30		CP/M	8080		200/25	LBA	
*XASM-48 Cross Assembler	1.30		CP/M	8080		200/25	LBA	
*XASM-65 Cross Assembler	1.95		CP/M	8080		200/25	LBA	
*XASM-68 Cross Assembler	1.96		CP/M	8080		200/25	LBA	
XMACRO-86 Cross Assembler	3.36		CP/M	8080		275/25	LBA	
XYBASIC Interpreter Extended	2.11		CP/M	8080		450/25	LBA	
XYBASIC Interpreter Extended CP/M	2.11		CP/M	8080		550/25	LBA	
XYBASIC Interpreter Extended COMP	2.0		CP/M	8080		450/25	LBA	
XYBASIC Interpreter Extended ROM	2.1		CP/M	8080		450/25	LBA	
XYBASIC Interpreter Integer	1.7		CP/M	8080		350/25	LBA	
XYBASIC Interpreter Integer COMP	2.0		CP/M	8080		350/25	LBA	
XYBASIC Interpreter Integer ROM	1.7		CP/M	8080		350/25	LBA	
Z80 Development Package	3.3	3.3	CP/M	Z80		95/20	LBA	
ZDT Z80 Debugger	1.41	1.41	CP/M	Z80		50/10	LBA	
ZSID Z80 Debugger	1.4		CP/M	Z80		130/15	LBA	

LBA=Lifeboat Associates

TIPS

This tip should be helpful to those who want to use BSTAM with the H89 Heath/Zenith CP/M by Magnolia. Two changes must be made to the file U8250/43.ASM in order to use this product when you have the abovementioned CP/M:

1-The line reading

```
BEGIN ORG 4303H
```

must be changed to:

```
BEGIN ORG 103H
```

2-The line near the end reading

```
CALL   4205H
```

with the comment, ";ANNOUNCE VERSION TO USER" must be altered to read:

```
CALL   05H
```

For users of Wordstar: a handy device makes it possible to incorporate words beginning with a "." in your texts--without fear that a line will not print should the "." fall at the beginning. Before and after the ".", ctl. PS (underscore) should be entered twice. On your screen, the line will look like this:

^S ^S. ^S ^SASM files are included....

The line, of course will print without the "^S"s and without underscoring.

This tip was contributed by Ward Christensen.

Version List Operating Systems

Asterisks (*) indicate new products or new versions.
12/15/80

Product Name	Version	Price	Available From
CP/M for Apple II with Microsoft BASIC	2.0	349	LBA
CP/M for Cromemco System 3 8"	1.4	145	LBA
*CP/M 2 for Durango F-85	2.22	170	LBA
CP/M for Heath H8 with H17 Disk	1.43	145	LBA
CP/M for Heath / Zenith H89	1.43	145	LBA
CP/M 2 by Magnolia for Heath / Zenith H89	2.2	250	LBA
CP/M for ICOM 3812	1.41	170	LBA
CP/M for ICOM 3712 with Altair Console	1.41	170	LBA
CP/M for ICOM 3712 with IMSAI Console	1.41	170	LBA
CP/M for ICOM Microfloppy (# 2411)	1.41	145	LBA
CP/M 2 for ICOM 4511/Pertec D3000 Hard Disk	2.22	375	LBA
CP/M for Intel MDS Single Density	1.4	145	LBA
CP/M 2 for Intel MDS Single Density	2.2	170	LBA
CP/M 2 for Intel MDS 800/230 Double Density	2.2	200	LBA
CP/M for MITS Altair 3202 Disk	1.41	145	LBA
CP/M for Micropolis Mod I - All Consoles	1.411	145	LBA
CP/M for Micropolis Mod II - All Consoles	1.42	145	LBA
CP/M 2 for Micropolis Mod I	2.20	200	LBA
CP/M 2 for Micropolis Mod II	2.20	200	LBA
CP/M for Compal Micropolis Mod II	1.4	145	LBA
CP/M for Black Hawk Micropolis Mod II	1.4	145	LBA
CP/M for Exidy Sorcerer Micropolis Mod I	1.42	145	LBA
CP/M for Exidy Sorcerer Micropolis Mod II	1.42	145	LBA
CP/M for NYLAC/REX Micropolis Mod II	1.4	145	LBA
CP/M for Vector MZ Micropolis Mod II	1.411	145	LBA
CP/M for Versatile 3B Micropolis Mod I	1.411	145	LBA
CP/M for Versatile 4 Micropolis Mod II	1.411	145	LBA
CP/M for Horizon North Star SD	1.41	145	LBA
CP/M 2 for Mostek MDX STD Bus	2.2	350	LBA
CP/M 2 for Ohio Scientific C3	2.22	200	LBA
CP/M for Sol North Star SD	1.41	145	LBA
CP/M for North Star SD IMSAI SIO Console	1.41	145	LBA
CP/M for North Star SD MITS SIO Console	1.41	145	LBA
CP/M for North Star DD	1.45	145	LBA
CP/M 2 for North Star DD/QD	2.21A	170	LBA
CP/M for Processor Technology Helios II	1.41	145	LBA
CP/M for TRS-80 5 1/4" (Model I)	1.41	145	LBA
CP/M 2 for TRS-80 Model II	2.24A	170	LBA
CP/M 2 for TRS-80 Model II + Corvus	2.24-C	250	LBA

Hard Disk Modules

Asterisks (*) indicate new products or new versions.
12/15/80

Product Name	Version	Price Alone	Price with CP/M 2.2	Available From
Corvus Module	1.6	125	80	LBA
KONAN Phoenix Drive	1.7	125	80	LBA
Pertec	1.6	125	80	LBA

LIFELINES®
1651 Third Avenue / New York, N.Y. 10028

NEW YORK N. Y.

FIRST CLASS MAIL
U. S. POSTAGE
PAID 28¢
Permit No. 416

FIRST CLASS MAIL