000, the problem is probably that there was no memory at that address.

If the WRITE data is zero and the read data NOT 377, the problem probably is that the memory slice was protected.

Elsewise, you may "trap" a bad memory slice by zeroing the last four locations (after making a note of them!) and doing the following changes:

-Change the START data to be your LAST memory location (like 017-377 in a 4K memory)

-Change the END data to be the beginning memory address to be tested (as in 000-050)

-Change location 000-024 to a DCX H (053).

. . . And run the program again. It will now stop, hopefully returning a new value in the ERROR locations. From these two addresses (the old address pointer you wrote down from locations 046 & 047, plus the new address pointer currently there) indicating between what two addresses (inclusive) that bad memory was found, you may have an indication that one of your IC's was bad (for instance, one bit would never go off: WRITE 000, READ 001 – it will usually be detected on WRITE 000.) and the memory pointers' difference will most likely be 1024. The memory slice and bit that is bad will indicate, with the help of a schematic, a bad memory chip.

Richard A. Kaapke 4485 Vision Dr., Apt. 9
KLUGES, Inc. San Diego, CA 92121

(All rights released to *DDJ* from Kaapke's Little Used but Greatly Esoteric Software, Incorporated)

REPRINTED *ALPHANUMERIC MUSIC WITH AMPLITUDE CONTROL* INCLUDES CORRECTIONS FROM AUTHOR

Malcom Wright has sent in three letters containing corrections and additions to his booklet on Altair computer music generation [see *DDJ*, Vol. 1, No. 5, for notes on this publication]. These letters (dated April 13, May 17, and June 7) have been included in the most recent reprint of the booklet, available from the PCC Bookstore.

COMPUTING CAREERS FOR DEAF PEOPLE

Proceedings of the 1975 ACM Conference on Computing Careers for Deaf People have been published by the Association for Computing Machinery. The Conference, sponsored by the ACM Special Interest Group on Computers and the Physically Handicapped (SIGCAPH), was held last April in the Washington, D.C. area. It featured 30 presentations (including 11 by deaf professionals) covering such topics as educational opportunities, special training programs, placement problems and solutions, federal legislation, on-the-job problems and solutions, and success factors.

Industry and government employers will find that these proceedings provide sound input to their plans for compliance with the requirements of the Rehabilitation Act of 1973 (Public Law 93-112).

The proceedings contain 125 pages and cost $6.75 for ACM Members and $9.00 for non-members. They are available, prepaid, from:

ACM Order Department
P. O. Box 12105
Church Street Station
New York, NY 10249

ERRATA TO ZEIGLER'S 6502 "BUG PROGRAM"

The March issue of *Dr. Dobb's Journal* [Vol. 1, No. 3] contained a "Breakpoint Routine for 6502's" submitted by John Zeigler. The final paragraph of the documentary text contains an error. It *should* read:

". . . it is necessary to load the interrupt vector, FFFE and FFFF, with *54* amd 02, respectively . . . "

PROPOSAL FOR *HANDY* SOFTWARE, WITH EXAMPLE

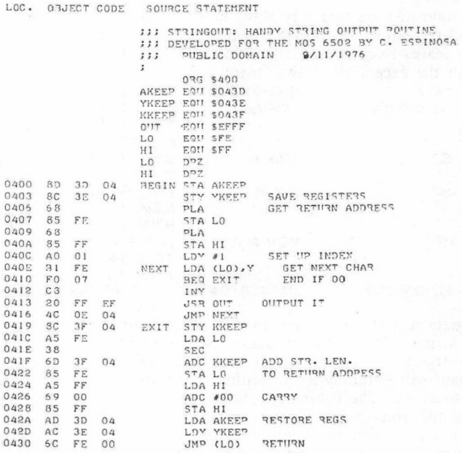
A STRING OUTPUT SUBROUTINE FOR THE 6502

Dear Jim: August 10, 1976

I have been noticing that in the *Journal*, the main subject has been large programs (BASIC's, monitors, text editors). I agree there is a need for large programs such as these, but I believe you should also concentrate on HANDY (Helpful Algorithms for Novice Do-It-Yourselfers) programs to save bytes in space-limited systems. I enclose my example: a string immediate-output subroutine for 6502-based systems. This routine saves pointers, loops, etc. normally used for string output by sequentially outputting the ASCII characters represented in hex in the bytes immediately following the Jump to Subroutine. After reading a terminating character (null), it returns to the instruction following the end of the string. No string addresses or lengths are needed.

The subroutine uses 40 (hex) contiguous bytes for program and intermediate storage and 2 zero-page bytes for indirect addressing. Calling the routine affects none of the registers, nor the stack. It has been implimented on an Apple Computer and copies of the program are being delivered to the Homebrew library and the CCC repository.

Chris Espinosa



```
LOC.  OBJECT CODE   SOURCE STATEMENT

                    ;;; STRINGOUT: HANDY STRING OUTPUT ROUTINE
                    ;;; DEVELOPED FOR THE MOS 6502 BY C. ESPINOSA
                    ;;;   PUBLIC DOMAIN    8/11/1976
                    ;
                          ORG $400
                    AKEEP EQU $043D
                    YKEEP EQU $043E
                    KKEEP EQU $043F
                    OUT   EQU $EFFF
                    LO    EQU $FE
                    HI    EQU $FF
                    LO    DPZ
                    HI    DPZ
0400  8D  3D  04    BEGIN STA AKEEP
0403  8C  3E  04          STY YKEEP   SAVE REGISTERS
0406  68                  PLA         GET RETURN ADDRESS
0407  85  FE              STA LO
0409  68                  PLA
040A  85  FF              STA HI
040C  A0  01              LDY #1      SET UP INDEX
040E  B1  FE        NEXT  LDA (LO),Y    GET NEXT CHAR
0410  F0  07              BEQ EXIT      END IF 00
0412  C3                  INY
0413  20  FF  EF          JSR OUT     OUTPUT IT
0416  4C  0E  04          JMP NEXT
0419  8C  3F  04    EXIT  STY KKEEP
041C  A5  FE              LDA LO
041E  38                  SEC
041F  6D  3F  04          ADC KKEEP  ADD STR. LEN.
0422  85  FE              STA LO     TO RETURN ADDRESS
0424  A5  FF              LDA HI
0426  69  00              ADC #00    CARRY
0428  85  FF              STA HI
042A  AD  3D  04          LDA AKEEP  RESTORE REGS
042D  AC  3E  04          LDY YKEEP
0430  6C  FE  00          JMP (LO)   RETURN
```