

# THE TARGET

November/December 1980

-- an AIM65 newsletter

There is a new kid on the block in the form of an enhanced AIM 65. It is the AIM 65/40. Its overall appearance reflects its roots in that it has a keyboard, display, and printer just like the AIM. The display and printer are 40 column units, however. Semi-graphics are supported also. The display and printer have their own individual 6502 processor. An intelligent printer will be one of its greatest improvements over the AIM 65/20. An improved memory map allows 48K of dynamic RAM, much the same languages, and a multi-file text editor are some of its features. The price will be approximately \$1700 and delivery will start during the second quarter of this year.

A number of subscriptions are running out with this issue. Stop now and send your renewal.

We are always looking for articles pertinent to the AIM 65. Remember that the size of an article does not determine its value; the content does. I want to emphasize that you should not send duplicate information to several publications. There is a chance that it could be published more than once and nobody gains from that.

1980 has brought increased information for the AIM 65. One significant item here is the Rockwell newsletter Interactive. Another is the user group support for particular hardware boards such as floppy disk controllers or video boards. In particular floppy disk user groups will provide an abundant source of programs. The problem here is that disks will not transfer from one system to another. 1981 will continue to bring information to the AIM 65 user.

A couple of weeks ago I spotted an AIM on TV and it was being used for data acquisition. Now that's what I call a real celebrity.

A program exchange has been started for the AIM 65. About 30 programs are available at this time but this number should change quickly. The exchange will be non-profit so costs involved should be minimal. Contact Jim Dantin, 1522 Springdale Drive, Owensboro, KY 42301.

I would like to thank numerous people for their assistance or contribution to the newsletter during the past year. Steve Bresson, Dick Buchen, Jim Butterfield, Mike Corder, Dave Cox, Leonard Crane, Mel Evans, Wendell Goring, Dick Grabowsky, Dale Hall, Larry Hollibaugh, Doug Kaynor, Knut Kvaal, Dale Lowery, Michael Rathbun, Eric Rehnke, Ron Riley, Steve Roberts, George Sellers, Steve Silber, Leo Scanlon, and John Wahlquist.

Copyright © 1980 Donald Clem Jr.

SOFTWARE	Music	Page 2
SOFTWARE	Memory Test	Page 9
SOFTWARE	Chess	Page 10
HARDWARE	Tape Recorder	Page 3
HARDWARE	Cheap RAM	Page 8
PRODUCTS	PL/65 Review	Page 4
	PL/65 Review	Page 5
	Memory-Mate	Page 6
	Review	

George Sellers  
 1033 Bishop Walsh Road  
 Cumberland, MD 21502

# Music

Enclosed are a couple of programs that may be of interest to new AIM 65 user's. The first(MUSIC) is a literal translation from the Kluge Harp program in KIM-1 User's Notes Vol. 1 Issue 3. The main item here is how to use the 6522 timer instead of the KIM-1 6530 timer. Two "songs" are included, "Happy Birthday" and "Theme from Close Encounters of the Third Kind". The notes and timing of these are approximately that of the Kluge Harp program, but may differ somewhat. The simple output device used is from the KIM-1 User's Manual.

The second program is a music subroutine (MUSRT) which can be called from BASIC. The BASIC driver (MUSRN) consists of generating a random note and random duration then calling the note to be played and at step 35 POKEs the note and duration into memory(5000-5FFF on my AIM). It takes a half an hour to fill up this 4K block. The sound is strange, something like the modern symphony music(ugh!).

-----  
 \*\*\* AIM/KIM/SYM \*\*\*

## <PROGRAMMABLE> RS-232 INTERFACE

FULLY PROGRAMMABLE  
 INCLUDING BAUD RATES !!!

Serial Echo Mode  
 Parity Generation & Detection  
 Programmable Interrupt  
 Parity, Framing and Overrun  
 Detection  
 Expansion Connector Provided  
 Chip Select is Jumper Selectable  
 Software Driver listings included

\$84.95 ASSEMBLED, 69.95 KIT

## FOBEL ENTERPRISES

DEPT. T, 552 E. EL MORADO  
 ONTARIO, CAL. 91764

-----

### MUSRT

```

==0000 PADD=$A003    ==0F00 START
==0000 PAD=$A001    A901 LDA #1
==0000 CRL0W=$EA13  8D03A0 STA PADD
==0000 RCHEK=$E907  A940 LDA #$40
==0000 RDRUB=$E95F  8D0BA0 STA ACR
==0000 RBYTE=$E3FD  ==0FOA NEW
==0000 HEX=$EA7D    A901 LDA #1
==0000 T1L=$A004    8552 STA COUN
==0000 T1H=$A005    ==0F0B LOOP2
==0000 IFR=$A00D    A553 LDA LENG1
==0000 ACR=$A00B    8551 STA LENG
==0000 TONE=$0050   ==0F12 LOOP1
==0000 LENG=$0051   A900 LDA #0
==0000 COUN=$0052   8D04A0 STA T1L
==0000 LENG1=$0053 A9AB LDA #$AB
==0000              8D05A0 STA T1H
                    ==0F1C LOOP
                    EE01A0 INC PAD
==010C              A650 LDX TONE
4C000F JMP START    ==0F21 LP1
==010F              CA DEX
                    D0FD BNE LP1
                    A940 LDA #%01000000
                    2CODA0 BIT IFR
                    F0F1 BEQ LOOP
  
```

### MUSRN

```

AD04A0 LDA T1L      5 N1=1:H=254:L=15
C651 DEC LENG      6 AT=80:AL=83
D0E0 BNE LOOP1     10 POKE4,0:POKE5,15
==0F32             15 FORAD=20480TO24574
C652 DEC COUN      STEP2
10D8 BPL LOOP2     20 N=INT(H*RND(N1)+N1)
A900 LDA #0        25 T=INT(L*RND(N1)+N1)
8D04A0 STA T1L     30 POKEAT,N:POKEAL,T
A9FF LDA #$FF      35 POKEAD,N:POKEAD+1,T
8D05A0 STA T1H     40 Y=USR(0)
A940 LDA #%01000000 45 NEXTAD
==0F42 WAIT
2CODA0 BIT IFR
F0FB BEQ WAIT
60 RTS
==0F48 END
.END
ERRORS= 0000
  
```

# Tape Recorder

## MUSIC

```

=0000 PADD=$A003      =OF1A LOOP1
=0000 PAD=$A001       A900 LDA #0
=0000 PHXY=$EB9E      8D04A0 STA T1L
=0000 PLXY=$EBAC      A9AB LDA # $AB
=0000 RCHEK=$E907     8D05A0 STA T1H
=0000 T1L=$A004       =OF24 LOOP
=0000 T1H=$A005       EE01A0 INC PAD
=0000 IFR=$A00D       A690 LDX TONE
=0000 ACR=$A00B       =OF29 LP1
=0000 TONE=$0090      CA DEX
=0000 LENG=$0091     D0FD BNE LP1
=0000 COUN=$0092     A940 LDA # %01000000
=0000 MEM=$0000       2C0DA0 BIT IFR
=0000                F0F1 BEQ LOOP
                    *=$010C
                    ADO4A0 LDA T1L
=010C                C691 DEC LENG
4C000F JMP START     DOE0 BNE LOOP1
=010F                =OF3A
                    C692 DEC COUN
                    10D7 BPL LOOP2
                    A900 LDA #0
                    8D04A0 STA T1L
                    A9FF LDA # $FF
                    8D05A0 STA T1H
                    A940 LDA # %01000000
                    =OF4A WAIT
                    2C0DA0 BIT IFR
                    F0FB BEQ WAIT
                    ADO4A0 LDA T1L
                    C8 INY
                    C8 INY
                    B90000 LDA MEM,Y
                    C900 CMP #0
                    DOB1 BNE NEW
                    =OF5B
                    209EEB JSR PHXY
                    2007E9 JSR RCHEK
                    20ACEB JSR PLXY
                    4COA0F JMP NEW
                    .END

```

## CLOSE ENCOUNTERS

```

(M)=0000 94 08 82 08
( ) 0004 A5 0A FD 04
( ) 0008 DB 10 00 01

```

## HAPPY BIRTHDAY

```

(M)=0000 C3 04 C3 04
( ) 0004 B2 08 C3 08
( ) 0008 94 08 9A 10
( ) 000C C3 04 C3 04
( ) 0010 B2 08 C3 08
( ) 0014 82 08 90 10
( ) 0018 C3 04 C3 04
( ) 001C 60 08 74 08
( ) 0020 92 08 9A 08
( ) 0024 B2 10 6D 04
( ) 0028 6D 04 74 08
( ) 002C 94 08 82 08
( ) 0030 94 10 00 00

```

Doug Kaynor  
308 NE 24th Ave  
Portland, OR 97232

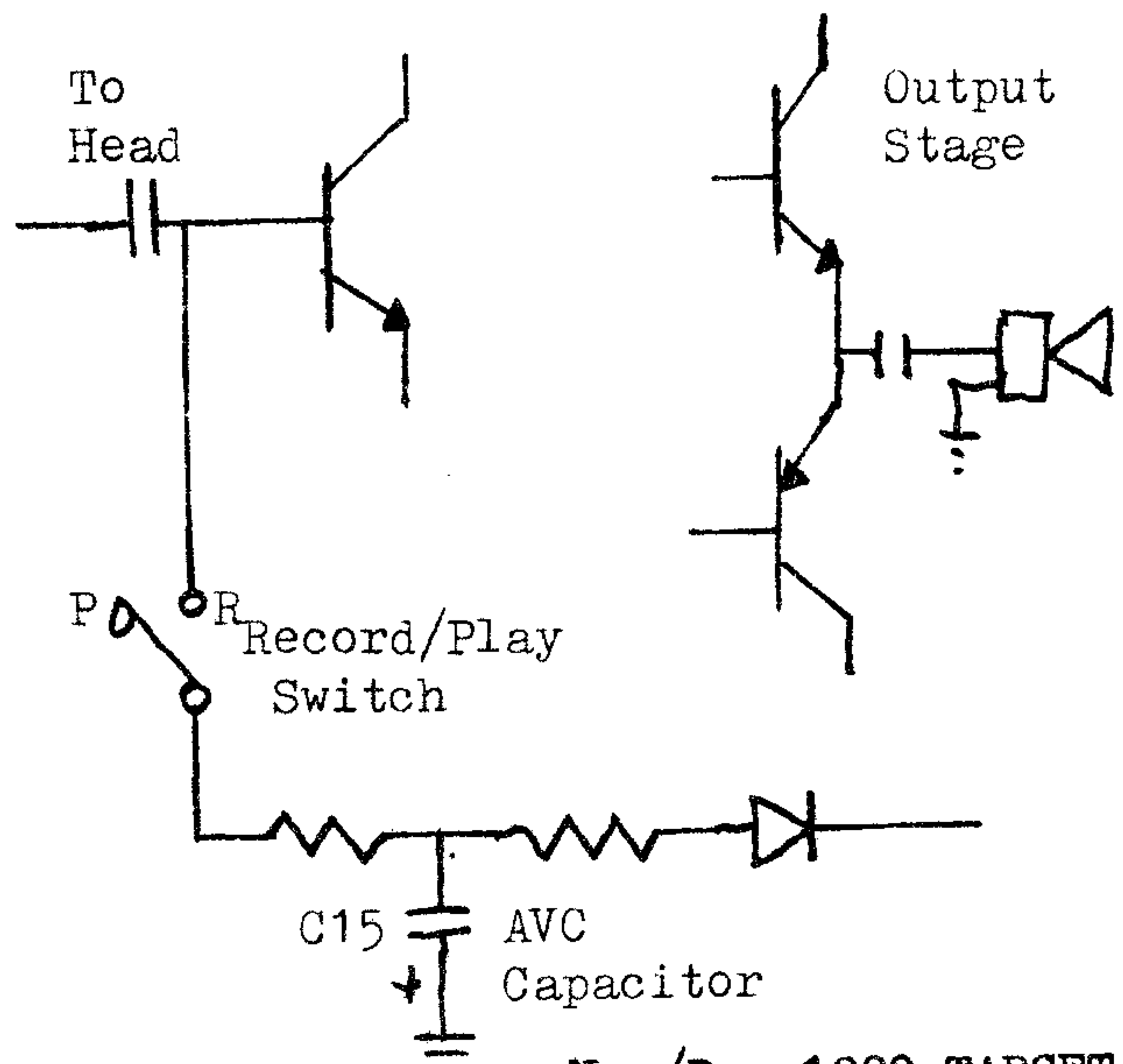
If you use a tape recorder for mass storage on your AIM, you have already discovered that you can use a smaller TGAP (\$A409) value if you don't use remote control. Here is another hint which will help if you find that you still can't record successfully with a TGAP of \$08.

On my AIM, I use a Realistic CTR-41 sold by Radio Shack. It is an excellent recorder for data storage except that it will not work unless TGAP equals \$13 or more. This is caused by an excessively long AVC time constant which causes the first one-half second or so of data to be distorted until AVC settles.

To fix the problem, I replaced C-15 which was a 22ufd @16volts with a 1ufd @16volts capacitor. Now I use a TGAP of \$08 or even less for text (object code may require a longer gap). I also installed a switch to add back in the larger capacitor for voice use. A 5 or 10ufd cap would probably be a good compromise for both data and voice use. Be sure to observe proper polarity when making the change. The same technique should work with any recorder.

Below is a typical setup. The AVC is a feedback network between output and input which is rectified and filtered and applied to the input when recording.

Write for help if needed. Please enclose a stamped, self addressed envelope for you return reply.



For a few months now, Rockwell has had available a new language for the AIM 65 called PL/65. This software is supplied in the form of two Read-Only Memory devices which plug into the sockets otherwise occupied by AIM BASIC. I took the plunge and purchased a set of PL/65 ROMs a few months ago, and have generally been satisfied with the performance of the product.

To begin with, though, let us establish what PL/65 is NOT:

1. It is not a replacement for BASIC for most casual or hobby users of the AIM. It is a compiler, rather than an interactive interpreter, meaning that there is an editing, compiling, and assembly process involved from user's code to finished product.

2. It is not a mathematically sophisticated language--anything beyond add, subtract, and single-byte compare must be implemented by the user as subroutines. You are more-or-less limited to the math capabilities of the AIM assembler in most cases.

It is, on the other hand, a good intermediate step between assembly language and higher-level languages like BASIC, in that it does a fine job with logical procedures like IF-THEN-ELSE, WHILE-DO, and FOR-TO-BY. In fact, if you are a structured-programming nut, you will find all the facilities here to enable you to write a program without a single GO TO statement. It is a whiz for programs which do a lot of comparisons, logical decisions, multi-byte moves, and multi-byte adds and subtracts. There are extensive facilities for defining and accessing single-dimension arrays; statements such as `IF L[G-B .AND RP] = A-Y THEN Q[PI] = M+Z` would take a bit of head-scratching and debugging to do in assembly language, unless you are one of those it-works-the-first-time geniuses.

I have found some drawbacks; chief among these is the manual. Although it covers the ground, you may have to look hard for the information you want. It is not at all up to the standards of the AIM User's Guide, or the BASIC manual. Another hassle is the lack of instructive error messages.

There is a good chance that the error(s) you have made will not produce an error message at all--the compiler may simply give up and execute a BRK instruction, leaving you wondering what happened. Unlike the AIM assembler, there is no errors-only listing mode, so if you don't have a CRT terminal, you may use up a lot of paper looking for the bomb that just went off during a compile.

Of course, learning any new programming language takes some work, and PL/65 is no exception. However, when you get used to the way it does things (and remember to end each statement with a semicolon) it can be rather rewarding. The final product is a machine language program, and therefore can have a considerable speed advantage over a similiar function in BASIC. The structure of the program which results is not extremely efficient, but this will be true of any compiler--especially one which has been squeezed into 8K of ROM. Meanwhile, if you have a logically complex programming project to undertake, PL/65 could very well cut a good deal of effort out of the picture. I have just completed work on a text processing system for the AIM which would undoubtedly be only half done now if I were working strictly with the assembler. The PL/65 source file is about 6K in size, but the intermediate assembler source file which the compiler generates is about 22K. Think how long it would take to type that stuff in...

I have found only one real bug (or undocumented limitation) so far: there is a limit to how deep you can nest WHILE-DO and IF-THEN-ELSE structures. If you exceed this limit (it is rather large) you will be in for much puzzlement, since the program will be legal in all respects, but will still explode during compile.

So, the final question: Do I think the expenditure of time and money on PL/65 was worth it? Yes, I do; for my applications, I have found it to be a big time and energy saver. You just have to get used to it. Michael Rathbun, PO Box 268, Kodiak, AK 99615

PL/65

Review by Ron Riley

I have had PL/65 for several months now and would like to make a few comments.

PL/65 is similiar to many other high-level languages. It isn't as easy to use as BASIC or FORTRAN and it doesn't have anywhere near as many features but it does generate machine language code in conjunction with an assembler.

Programs are produced 3 or 4 times faster than when using assembly language techniques. PL/65 generated code is 1.5 to 2 times longer than well done assembly language. Development time is traded off against code efficiency. The user has the option of optimizing the code prior to assembly.

The chip set is \$125 and is available from:Compas Microsystems, 224 S.E. 16th Street, Ames, IA 50010; Excert,Inc, PO Box 8600, White Bear Lake, MN 55110.

MINI T is a miniature text processor. It justifies margins, centers text, and many other things. Operates on AIM editor text files with single letter commands. Requires 4K and printer. Object: \$30.00 Manual alone: \$5.00 (deductable).

DSASM: a 6502 disassembler which produces tape source file from object code in memory or tape; has option for table of meaningful labels; creates file of label definitions; segments source into logical blocks. Requires 4K. Object: \$25.00 Manual only: \$3.50

MACRO processes AIM editor files to give you macro-assembler capability. Can be used to cross-assemble for other CPUs. Object: \$15.00

Catalog of other products now available.

POLAR SOLUTIONS  
Box 268  
Kodiak, Alaska 99615

## Power to your AIM

### CORRECTIONS--Touch Tone Dialer

There are two errors in the listing. Under the heading "Monitor Equates", PLS1 should be PSL1.

The second error is two missing instructions following the label TONES. The correct listing should be:

### TONES INY

```
LDA (NOWLN),Y ;NEXT # TO BE DIALED..  
CMP #D ;UNTIL END OF LINE..  
BEQ EXIT  
CMP #20 ;OR "SPACE" ENCOUNTERED  
BEQ EXIT  
JSR HEX ;CONVERT TO HEX
```

The missing instructions are underlined. The program will work as published. These two instructions were there to allow the user to end his "number" with a space, and then add comments in the same line- which would be ignored by the dialing program. The text does not specifically point out this feature, but it is alluded to in the next to last sentence of the fourth program.

Treat your AIM to quality power supply:

1. Designed to Rockwell's specifications for the AIM 65 (5 volts at 2 amps, regulated; 24 volts, .5 amps avg, 2.5 peak, unregulated.)
2. Overvoltage protection to protect the expensive circuits in your AIM. (5 volt output)
3. Handsome all metal case (two tone blue).
4. Fuse (externally accessible), switch, pilot light, line cord, cable from power supply to AIM-all included.
5. Conservative thermal design for long life.
6. Satisfaction or your money back.

\$64.95 plus shipping (5 lbs.)      CompuTech  
CA residents add 6% sales tax      Box 20054  
VISA/MC, cashier's check, or      Riverside,  
registered check, personal      CA 92506  
check (allow 2 wks. to clear)

Nov/Dec 1980 TARGET 5

# Memory-Mate Review

Memory-Mate from Forethought Products  
87070 Dukhobar Rd., Eugene, OR 97402.

The Memory-Mate has several features with the primary item being a memory board. In addition it has two 6522 VIA peripheral chips and sockets for up to 4K of EPROM. The Memory-Mate in conjunction with the STD-Mate board along with an AIM-65 will substitute for a STD BUS processor card.

The memory chips used are dynamic and therefore consume very little power. The board may be populated with either 4115, 4215, or 4116 RAMs to a maximum of 48K of endless (?) RAM.

A study of Memory-Mate would be incomplete without mention of the parity select option. Parity checking and indication allow immediate annunciation of faulty memory. Fault indication may be through a LED or through an interrupt. A jumper option enables the interrupt to be serviced through normal polling. Some slight initialization is required to bring parity check on-line. In addition parity is enabled or disabled by a short write routine.

Write protection is enabled in much the same way as parity. It uses a LED also and may generate an interrupt if desired. Several examples are provided in the manual supporting the need for write protected memory. Memory may be write protected in 4K blocks.

The two supplied VIAs are completely available for the Memory-Mate user. The signals are available at an edge connector. The manual suggests an I/O connector arrangement that will allow two DB25 connectors to interface peripherals. There are a multitude of uses for these I/O lines. The reset lines on the 6522s as supplied are jumpered for power-on reset only. This allows setting and forgetting of the 6522 without worrying about reset re-initialising the chip.

The EPROM sockets support the use of 2716 (Intel), TMS2508, TMS2516, TMS27580L0 or L1 EPROMS. The sockets can be jumpered to 4K areas of memory.

An amplifier with speaker output is provided on-board so no external amp is required. A square wave is output by simply writing to a particular memory slot. A speaker line is toggled on every write so programming of the output is quite simple. Routines

have been provided to generate several sounds. The user may easily generate his own or even expand on these. The program shown here in conjunction with the supplied noise routines will provide an end-of-line indicator for use with the AIM editor.

Hardware is provided to mount the Memory-Mate directly under the AIM. This places the AIM board at the top of the AIM keyboard. The Memory-Mate power requirements are 5V at 1A and 12V at 200 ma.

The manual supplied is excellent and provides the software to enable and maintain the boards features. The manual is available seperately for \$7.50.

The Memory-Mate is a second generation board with supporting software with a reasonable price. It may be just the item to solve all your expansion needs.

---

The Forethought Products AIM-Mate series of expansion products look to be an interesting approach to AIM expansion. Memory-Mate, STD-Mate, and Power-Mate are available now with a card cage, case, Video-Mate, and Floppy-Mate coming in the near future. Donald Clem

---

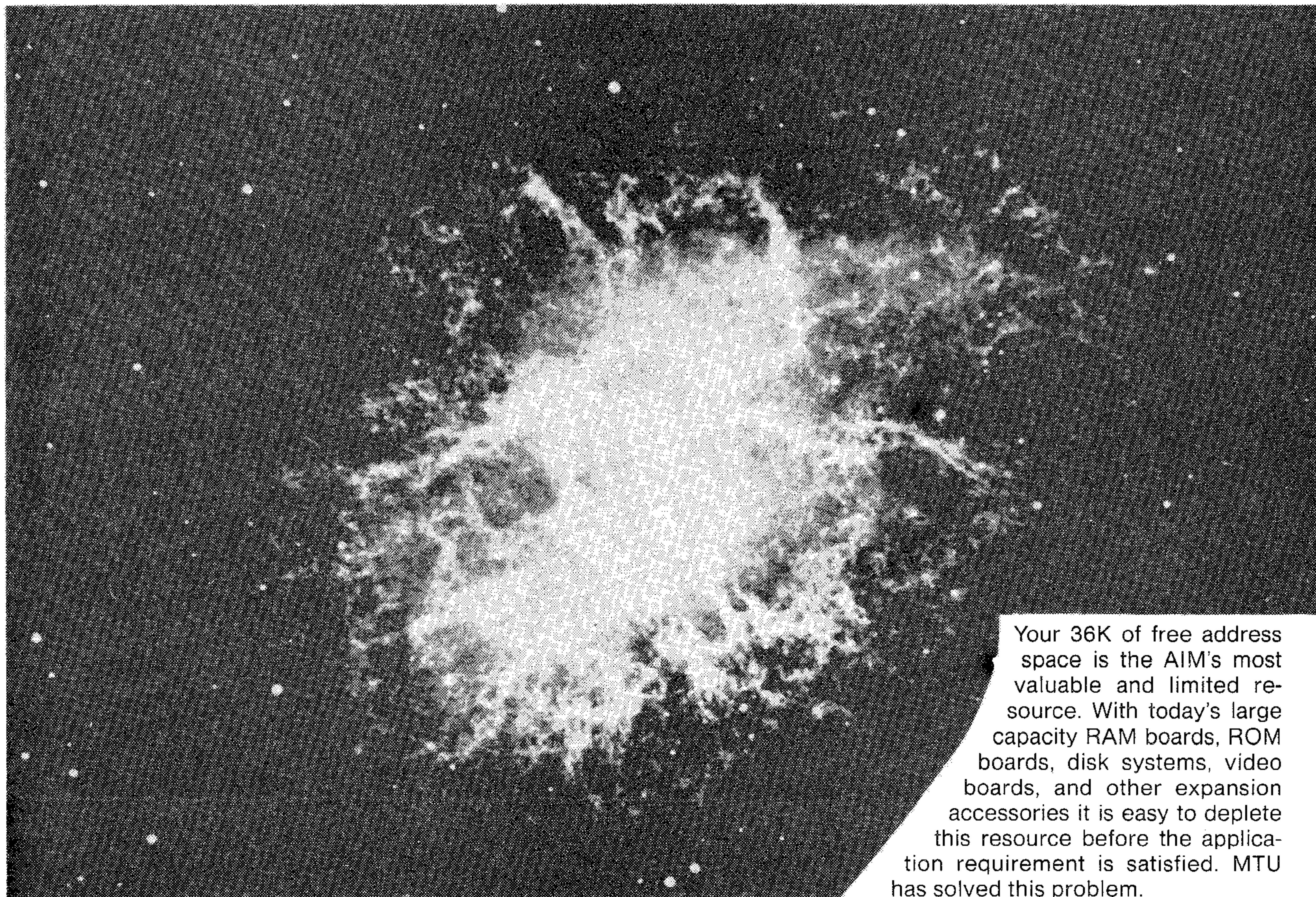
## END-OF-LINE INDICATOR

F1-enables  
F2-disables

(M)=010C 4C 90 02 4C  
( )=0110 B0 02

0290 A9 LDA #02  
0292 8D STA A407  
0295 A9 LDA #A0  
0297 8D STA A406  
029A 4C JMP E182  
029D EA NOP  
029E EA NOP  
029F EA NOP  
02A0 48 PHA  
02A1 AD LDA A415  
02A4 C9 CMP #37  
02A6 D0 BNE 02AB  
02A8 20 JSR 0227  
02AB 68 PLA  
02AC 4C JMP EF05  
02AF EA NOP  
02B0 A9 LDA #EF  
02B2 8D STA A407  
02B5 A9 LDA #05  
02B7 8D STA A406  
02BA 4C JMP E182

## A BRILLIANT FUTURE FOR YOUR AIM-65 WITH THE BANKER MEMORY™



Your 36K of free address space is the AIM's most valuable and limited resource. With today's large capacity RAM boards, ROM boards, disk systems, video boards, and other expansion accessories it is easy to deplete this resource before the application requirement is satisfied. MTU has solved this problem.

THE BANKER MEMORY contains 32K of RAM, 4 PROM sockets for 2716/2732/2332, a PROM programmer, 40 bits of parallel I/O, and 4 timers from two 6522 I/O chips. Addressing is extremely flexible with the RAM independently addressable in 4K blocks, PROM's independently addressable, and I/O addressable anywhere on a 64 byte boundary (even in AIM's I/O area at AXXX by adding a single jumper to the AIM).

This may sound familiar, but read on! Unlike other AIM compatible memory boards, THE BANKER MEMORY has on-board bank-switching logic! The four 8K blocks of RAM plus the 4 PROM sockets make up 8 **resources**, each associated with a bit in an Enable Register. Through this Enable Register resources may be turned on and off under software control. When a resource is off, its address space is freed for other uses. You can even put BANKER resources at the same address and switch among them for virtually unlimited RAM and PROM expansion! You can even have multiple page zero's and stacks! Do you need 160K byte of memory? It only takes 5 of THE BANKER MEMORY boards and you end up with 5 page zeros and stacks to boot!

There's more! The BANKER MEMORY also incorporates 18 bit addressing which allows for the 256K address spaces of the future. RAM, PROM, and I/O each has its own full 18 bit address decoder which allows these resources to be in different 64K banks. This board and other MTU products, such as our 320 by 200 dot VISIBLE MEMORY and Floppy Disk Controller with 16K DMA RAM, can turn your AIM into a truly powerful 6502 computer that far surpasses the packaged systems in functional performance.

INTRODUCTORY SPECIAL K-1032-1 32K BANKER MEMORY FULLY ASSEMBLED AND TESTED \$395.00 (\$450.00 as of March 1, 1980) or the K-1032-2 16K RAM only with bank switching and 18 bit address bus only \$295.00

Isn't it time you took a closer look at MTU — we offer you power now with an eye to the future.

WRITE OR CALL TODAY FOR OUR 48 PAGE FALL 1980 6502 CATALOG

International requests include \$1.00

VISA and MASTERCARD accepted

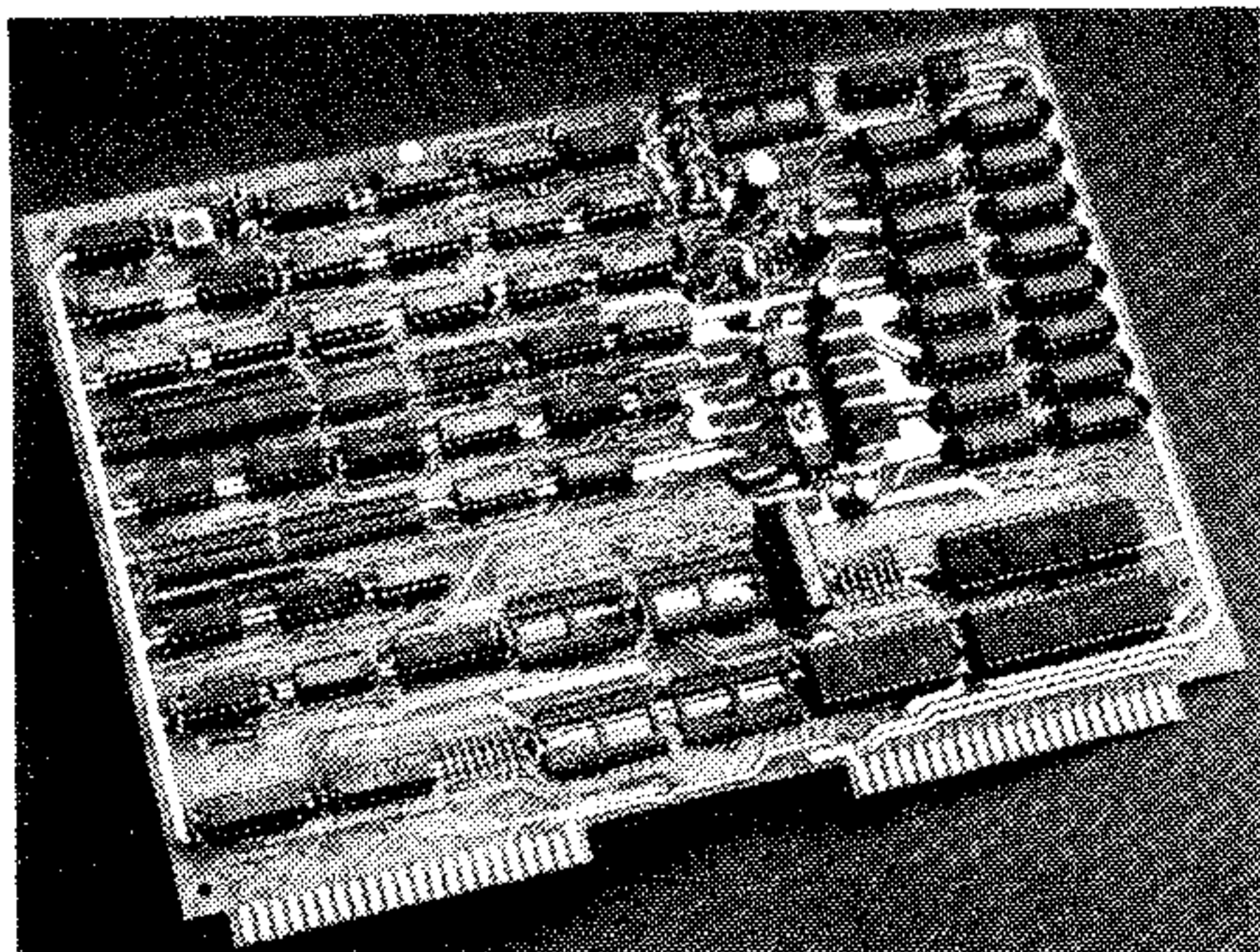


Photo credit:  
SUPERNOVA CRAB NEBULA:  
Palomar Observatory,  
California Institute of Technology

**MTU**  
**Micro Technology Unlimited**  
2806 Hillsborough Street  
P.O. Box 12106  
Raleigh, NC 27605, U.S.A.  
(919) 833-1458

# Cheap RAM

The California Computer Systems CCS XVI, model 2016B, is a 16K static RAM module utilizing 2114's and is designed for S-100 signal standards. It may be plugged directly into AM100 and Cromemco systems.

Conversion to the AIM 65 is simply a matter of wiring the S-100 connector, being careful to observe the proper signal-pin relationship as some connector numbering systems vary, to the 50 pin edge connector. The DI-D0 signals for the memory board must be tied together and routed to the like data bus of the AIM expansion interface. The only variation is that the AIM R/W pin W must be used for the memory's MWRITE, pin 66. All unused terminals may be left open, or if you desire tie all inputs high through a pull-up. Fire it up!

If you want to be frugal, eight DIP IC's, four SIP resistor networks, three DIP switches, and one LED or two LED's may be cast out without affecting performance. The IC's are in sockets and are:

U12,U13	74LS175 Quad D-register
U14	74LS08 Quad 2 input nand
U11,U15,U16	74LS30 8 input nand
U52	75453 Dual OC peripheral driver
Z4,5,6,7	7X2.7&4.7K resistor network
S5,S7	8 position DIP rocker switch
S6	4 position DIP rocker switch
DS1,2	LED

If your 5 volt logic can supply 2A then four T0220 voltage regulators are gained.

Not a bad spare parts list!! But wait, even without the above the following spares exist:

U53 pins 3,4,5,6	U59 pins 3,4,5,6,9,10
U55 pins 11,10,13,12	11,8
U56 pins 12,11,14,13	U60 pins 1,13,11,9,10
	8
	U61 pins 8,9,10,11,12
	3

One jumper is needed. In U14 pins 2-3 must be jumpered (socketed). If you wish to maintain the Board Select LED functional jumper U53 pin 6 to U11 pin 3. S1 thru S4 select the starting address individually for each of the 4K blocks.

For continuous memory set them to \$1000, \$2000, \$3000, \$4000, in any order. You may want to partition sections for BASIC, TEXT EDITOR, and ASSEMBLY PATCHES to allow default values at the high ends for convenience.

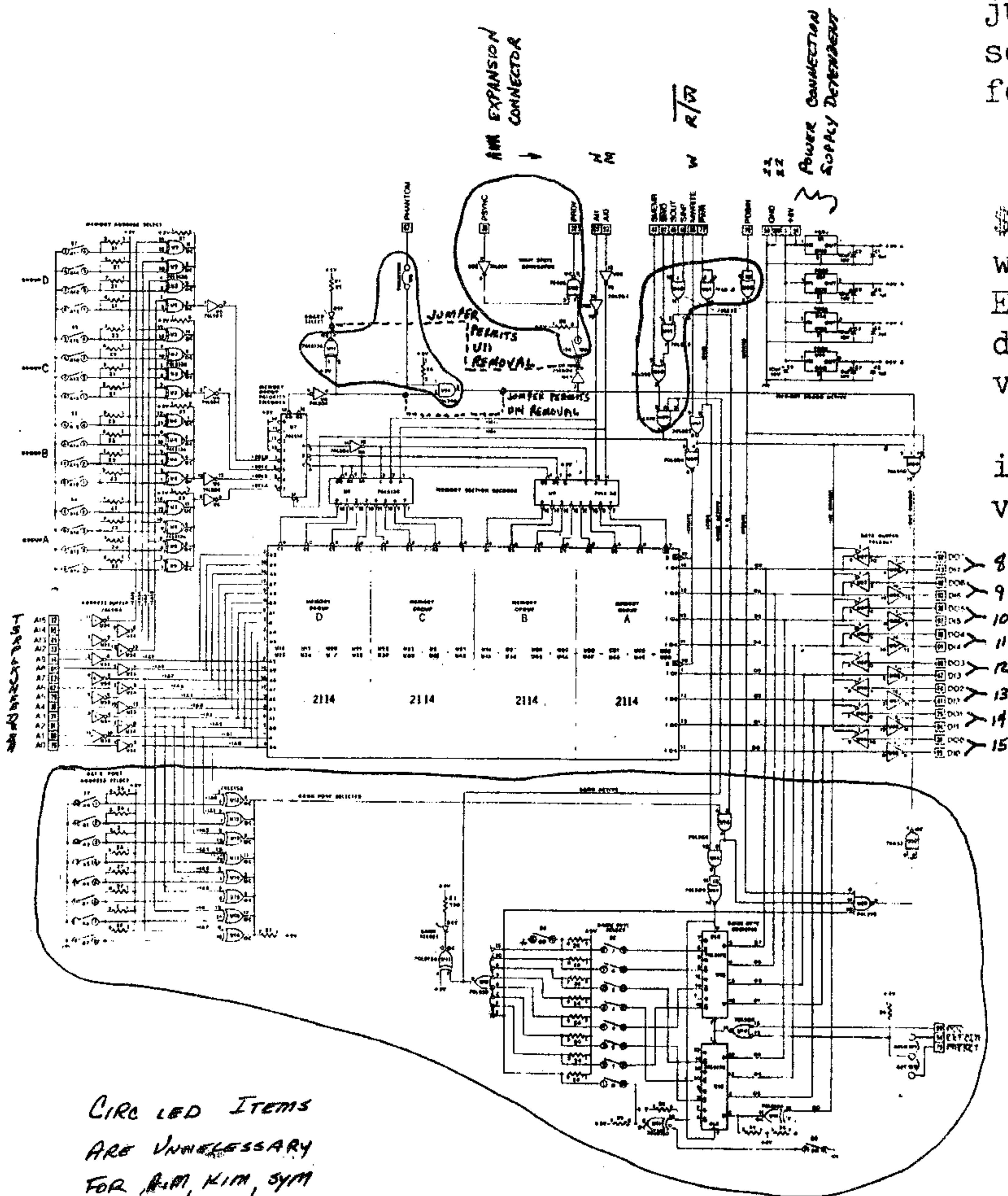
See the diagram for more details. Also included is a memory test to check the conversion out.

## GENERAL INFORMATION

Article contributions are always welcome. Program listings may or may not be retyped. When submitting information on AIM thermal paper adjust the darkness control to its darkest setting. Artwork will not be redrawn so please submit your best work. Artwork may be oversize if necessary and will be reduced to proper size.

Text should accompany articles to explain what is being done, how it is done, and how it may be modified to suit the user.

Please submit a self addressed stamped envelope for any replies that you desire.





=====
   
DYNAMIC MEMORY TEST
   
=====

"F1"<D>FROM=XXXX
   
          TO=XXXX
   
CR OR SP BEGINS TEST

"F2"<J>FROM=LAST ENT
   
          TO=XXXX
   
CR OR SP BEGINS TEST

"F3"<I>FROM=LAST ENT
   
          TO=LAST ENT
   
STARTS IMMEDIATELY

- - - - -
   
DISPLAY COUNTS # OF
   
TIMES ENTIRE MEMORY
   
HAS PASSED A WRITE
   
THEN READ COMPARISON
   
+ COMPLEMENT PATTERN
   
CAPACITY=\$FFFF
   
=65536 CYCLES PASSED

ALL FUNCTIONS:
   
ANYKEY=PAUSE
   
SPACEBAR=ALTERNATE
   
HOLD DOWN ESC TO EXIT

FAILURE TERMINATES
   
TEST.  DISPL=MSG+LOC
   
<MEM FAIL XXXX

CONTENTS OF MEM ADR
   
LSB \$0A & ITS CPLMNT
   
ARE COMPARED WITH EA
   
MEM LOC MAKING TWO
   
TESTS/LOC CHANGING
   
PATTERN FOR ADJACENT
   
MEM LOCS, REPEATING
   
AFTER 256=\$FF
   
PATTERN= FF FE FD FC
   
          FB . . . . 02 01 00 FF

=====
   
PATTERN VARIATIONS:

COMPLMT CHECKERBOARD
   
\$3A= LDA #AA
   
1= 10101010
   
2= 01010101 MEM=\$55

00'S IN ALL MEM
   
STORE COMPL #00=\$FF
   
\$3A= LDA #FF
   
1= 11111111
   
2= 00000000 MEM=\$00

=====
   
DLH REV 5/80

FROM=A409 TO=A409
   
OUT=T F=XDMT3 T=1
   
MORE?Y
   
FROM=10C TO=114
   
MORE?Y
   
FROM=10 TO=84
   
MORE?N

<I>FROM=200 TO=4FFF
   
0001
   
0002
   
0003
   
0004
   
0005
   
0006
   
0007

<J>TO=4000
   
0001
   
0002
   
0003
   
0004

<I>
   
0001
   
0002
   
0003
   
0004

<J>TO=5000
   
MEM FAIL 5000

<M>=200 FF FE FD FC
   
< > 0204 FB FA F9 F8
   
< > 0208 F7 F6 F5 F4
   
< > 020C F3 F2 F1 F0
   
< > 0210 EF EE ED EC

<I>
   
003A A9 LDA #AA

<I>
   
MEM FAIL 5000

<M>=200 55 55 55 55
   
< > 0204 55 55 55 55
   
< > 0208 55 55 55 55
   
< > 020C 55 55 55 55

<I>
   
003A A9 LDA #FF

<M>=200 55 55 55 55
   
<I>TO=
   
<I>
   
MEM FAIL 5000

<M>=200 00 00 00 00
   
< > 0204 00 00 00 00
   
< > 0208 00 00 00 00
   
< > 020C 00 00 00 00

MEMORY TEST in zero page

< > A409 20 04 0A 06 by Dale Hall

<K>\*=10C
   
/03
   
010C 4C JMP 0010
   
010F 4C JMP 001D
   
0112 4C JMP 002A

<K>\*=10
   
/51

0010 20 JSR E7A3
   
0013 AD LDA A41C
   
0016 85 STA 0E
   
0018 AD LDA A41D
   
001B 85 STA 0F
   
001D 20 JSR E7A7
   
0020 AD LDA A41C
   
0023 85 STA 0C
   
0025 AD LDA A41D
   
0028 85 STA 0D
   
002A A9 LDA #00
   
002C 85 STA 00
   
002E 85 STA 01
   
0030 A0 LDY #00
   
0032 A5 LDA 0E
   
0034 85 STA 0A
   
0036 A5 LDA 0F
   
0038 85 STA 0B
   
003A A5 LDA 0A
   
003C 91 STA (0A),Y
   
003E D1 CMP (0A),Y
   
0040 D0 BNE 0078
   
0042 49 EOR #FF
   
0044 91 STA (0A),Y
   
0046 D1 CMP (0A),Y
   
0048 D0 BNE 0078
   
004A A5 LDA 0B
   
004C C5 CMP 0D
   
004E D0 BNE 006F
   
0050 A5 LDA 0A
   
0052 C5 CMP 0C
   
0054 D0 BNE 006F
   
0056 E6 INC 01
   
0058 D0 BNE 005C
   
005A E6 INC 00
   
005C 20 JSR EA13
   
005F A5 LDA 00
   
0061 20 JSR EA46
   
0064 A5 LDA 01
   
0066 20 JSR EA46
   
0069 20 JSR E907
   
006C 4C JMP 0030
   
006F E6 INC 0A
   
0071 D0 BNE 003A
   
0073 E6 INC 0B
   
0075 4C JMP 003A
   
0078 A5 LDA 0B
   
007A 8D STA A41D
   
007D A5 LDA 0A
   
007F 8D STA A41C
   
0082 4C JMP EB33

11/79
   
Dale L. Hall
   
19904 Grevelink Ave.
   
Torrance, Calif. 90503

FROM
   
ADR-L
   
FRADR-L
   
ADR-H
   
FRADR-H
   
TO
   
ADR-L
   
TOADR-L
   
ADR-H
   
TOADR-H
   
A=0 Reset
   
MSD cycle
   
LSD counter
   
Y index = 0
   
[FRADR-L]
   
Temporary FRADR-L T
   
[FRADR-H]
   
Temporary FRADR-H T
   
Test pattern
   
Store TP [0B0A]
   
[0B0A] = A reg ??
   
Branch on fault
   
Test patt complemen
   
Store CTP [0B0A]
   
[0B0A] = A reg ??
   
Branch on fault
   
TADR-H
   
TOADR-H Last MSD ?
   
Branch if 0D ≠ 00
   
TADR-L
   
TOADR-L Last LSD ?
   
Branch if 0D ≠ 00
   
LSD = LSD+1
   
Branch if 01 ≠ 00
   
MSD = MSD+1
   
CRLOW Clear display
   
Cycle counter MSD
   
NUMA Display = MSD
   
Cycle counter LSD
   
NUMA Display = LSD
   
RCHEK Keyboard scan
   
Next cycle
   
TADR-L = TADR-L+1
   
Branch if 0A ≠ 00
   
TADR-H = TADR-H+1
   
Test next mem loc
   
A=FAILED MEM LOC-H
   
ADR-H = TADR-H
   
A=FAILED MEM LOC-L
   
ADR-L = TADR-L
   
MEMERR MSG+ADR -L

Steve Silber  
3904 Old Oaks Dr  
Bryan, TX 77801

# Chess

I have written some input and output routines for Peter Jennings' Microchess program and integrated them into Microchess. The routines allow for input of moves in a fairly normal type of notation (k2-k4 for normal opening P-K4) and also generates a list of moves on the printer for a permanent record of the game. In addition, a routine is provided which will allow for the examination and changing of the board position. This is very useful to verify the board during a game, or to set up an end game or game in progress. The source program consists of five files, each of which will load into a 4K system. The object code loads from \$200 to about \$900, in addition to all of page zero. No code resides on page one.

If anyone is interested in a copy of this program, I am willing to make a copy of the source files and assembled object files as well as the source listing (approximately 28 pages) if they will send me a blank cassette tape and \$5.00. If there is no interest in the copy of the source listing, then I can supply just the five source files and the object file for \$2.00 plus a blank cassette. Please be sure to specify what tape gap you normally use for source files.

10 TARGET Nov/Dec 1980

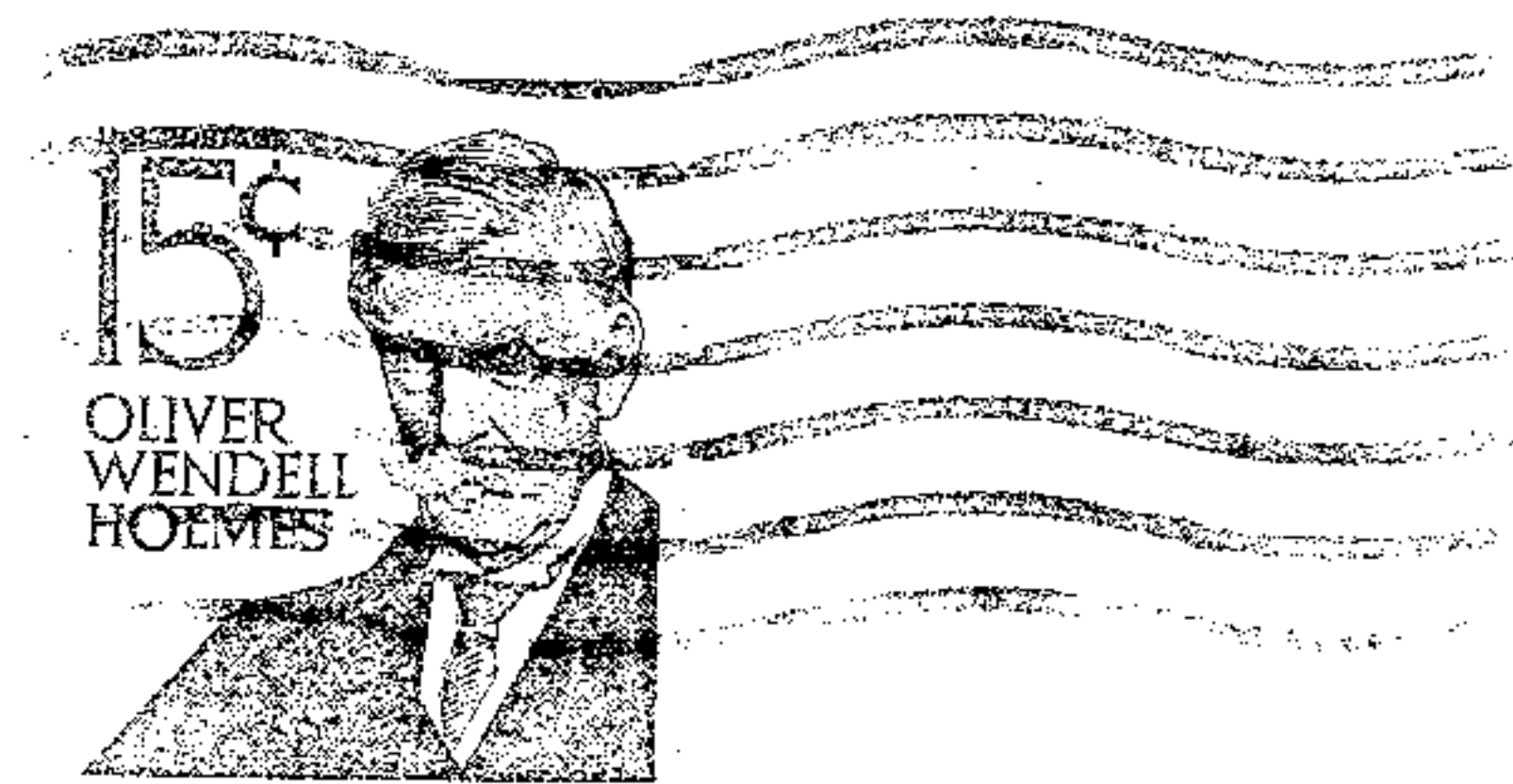
THE TARGET  
c/o DONALD CLEM  
R.R. #2, CONANT RD.  
SPENCERVILLE, OHIO 45887

Back Issues- A consolidated 1979 issue is available for \$6.00 (\$12). In addition 1980 issues are available beginning with the January/ February and at subsequent two month intervals. Individual 1980 issues are \$1.00 (US and CAN, \$2.00 elsewhere).

Time to Renew- The mailing label contains the last issue that you will receive. If no date appears you have at least two issues left on you subscription.

Target- an AIM 65 newsletter is published bimonthly with an annual subscription rate of \$6.00 in the US and Canada, \$12.00 elsewhere. Contact Donald Clem, RR# 2, Spencerville, OH 45887.

6502 Assembly Language Programming by Lance A. Leventhal from Osborne/McGraw-Hill. \$12.50 plus \$1.00 handling (non-US or CAN-\$4.00) C/O Donald Clem



Excert Inc  
PO Box 8600  
White Bear Lake, MN 55110