THE
SYM/KIM APPENDIX
TO THE
FIRST BOOK OF KIM*

by
Robert A. Peck

* Published by Hayden Book Company

Introduction

Since the SYM-1 is a relative newcomer, there is not much software provided for it. However the KIM-1 has been around somewhat longer and has been the subject of many construction and programming articles.

As a learning exercise, game programs often provide the user with a good introduction to the capabilities of a system. Therefore this appendix was written to allow the set of games from the FIRST BOOK OF KIM to execute directly on the SYM-1.

All programs were designed to run orignially on a 1K system. The basic SYM-1 is also only 1K so the prime goal of the translation was to keep the program size within the bounds of a basic SYM-1.

A secondary goal, which has been followed, was to keep most of the KIM program intact. This allows the user to follow the logic and instruction generated in the book and yet see where the changes were required and how they were accomplished.

This appendix contains only the program lines which require changes, thus it does not contain any complete programs. It is intended to supplement the FIRST BOOK OF KIM so it is assumed that the user already has access to this book.

If you have any questions, problems or suggestions for this or future manuals, you may contact me at the address below. If you wish an answer please include a self-addressed stamped envelope.

Even with best efforts, typographical errors will occur, but I hope that this pamphlet will serve to help you to understand some of the ways to adapt KIM programs to run on your SYM-1.

Robert Peck
P.O. Box 2231
Sunnyvale, CA 94087

RECREATIONAL PROGRAM CONVERSIONS

The following programs have been converted and are a part of this volume:

ADDITION
ASTEROID
BAGELS
BANDIT
BITZ
BLACKJACK
BLACK MATCH
CARD DEALER
CHESS CLOCK
CODE TEST
CRAPS
DUEL
FARMER BROWN
HI-LO
HORSERACE
KEYTRAIN
KIM NIM
KIM-TAC-TOE
LUNAR LANDER
MULTI-MAZE
PING PONG
QUICK
REVERSE
TEASER
WUMPUS

MUSIC BOX requires an external speaker - only programs which can run on a stock, basic SYM-1 were translated here. CLOCK and TIMER can be done more accurately using the 6522 on-board timers. Since this volume was intended to contain no "complete" programs, these programs were left out.

Typographical Errors/Omissions

To aid in understanding the KIM programs, I felt it best to include a set of corrections or additions to the book. These are outlined below and apply to my copy (2nd printing, 1978). They may have been corrected since then but are provided here for the users convenience.

P.27 ASTEROID, LINE Ø269; should be labeled NOBT.

P.42 BLACKJACK,LINE Ø292; should be labeled HIT.

P.90 PING PONG,HEX DUMP; Ø325 should be "Ø8" not "CØ".

P.108 WUMPUS,LINE Ø36C; DELETE DUPLICATE LABEL "ROOM".

Programming Notes - General

In the pages which follow, an astute programmer may find certain areas where changes may not have been added or where certain functions could possibly have been done more simply or elegantly. A super-efficient method was not the object of the exercise.

Rather the main object was to make maximum use (where appropriate) of the SYM-1 monitor and to provide useful game software during this learning experience.

Notes on Program Conversions

The basic SYM-1 has RAM space available from ØØØØ to Ø3FF. We can fit programs in without restrictions from Ø2Ø0-Ø3FF. The other areas however are slightly restricted.

The monitor program uses the area from approximately Ø1D1-Ø1FF for the stack so if any of your program or data is written here it may be destroyed when you hit the reset key. Also note that the more levels of subroutine nesting which is used, the further the stack will extend downward from Ø1FF into the memory space. So some care must be used to avoid wiping out your data.

The monitor also uses locations ØØFE and ØØFF. If you try to use the monitor routines and also modify these locations, the program will get lost because it stores program counter data there.

The SUPERMON memory map specifies that the monitor uses all ØØFØ to ØØFF. On some occasions then, when simulating the KIM output routines, I have substituted A9, AA, and AB or E9, EA and EB for F9, FA and FB. But it appears in most cases (at least for these programs) the original use of F9, FA, FB locations does not affect the program operation.

All of the programs require the use of the keyboard and display. Since the registers which control both are in the write-protected area of memory, the first instructions in each program is always: 2Ø 86 8B JSR ACCESS to allow the program to write into system RAM.

Certain of the routines which will be seen often in one form or another throughout this text are the following:

SCAND2 - A Subtitute for KIM SCAND

GETKEY2 - A substitute for KIM GETKEY

These routines are presented here in their most general format for easy reference:

```
D8          SCAND2  CLD             ;CLR Dec (Need if called when in Dec)
A9 2B               LDAIM   #$2B    ;LOAD ACC with Hex 2B (setup no
8D 7Ø A6            STA     A67Ø    ;   scan for movement into DISBUF)
A5 FB               LDA     ØØFB    ;LOAD ACC with contents of ØØFB
2Ø FA 82            JSR     OUTBYT  ;MOVE two hex char into DISBUF
A5 FA               LDA     ØØFA    ;LOAD ACC with contents of ØØFA
2Ø FA 82            JSR     OUTBYT  ;PUSH two hex char into DISBUF
                                    ;   (Moves Prev)
A5 F9               LDA     ØØF9    ;LOAD ACC with contents of ØØF9
2Ø FA 82            Jsr     OUTBYT  ;PUSH two hex char into DISBUF
                                      (Moves Prev)
2Ø Ø6 89            JSR     SCAND   ;JUMP to SYM SCAND routine
A9 Ø6               LDAIM   #$Ø9    ;LOAD ACC hex Ø
8D 7Ø A6            STA     A67Ø    ;RRESTORE the SCANVEC to original
F8                  SED             ;SET DEC(only if in DEC when
                                    ;   SCAND2 called)
6Ø                  RTS             ;RETURN

D8          GETKEY2 CLD             ;CLEAR DEC (If in DEC when called)
2Ø 88 81            JSR     SAVER   ;JUMP to register-SAVE routine
2Ø 23 89            JSR     KEYQ    ;See if key down. Read input port
2Ø 2C 89            JSR     LRNKEY  ;Bring ASCII of key into ACC,
                                    ;   no dis scan
2Ø 75 82            JSR     ASCNIB  ;Convert ASCII to hex in low nibble
                                    ;   of ACC
F8                  SED             ;SETDECmode(IfenteredwithDECSET)
4C B8 81            JMP     RESXAF  ;RESTORE regs except ACC and
                                    ;   Cond CODE/RTS
```

Some explanation is in order for these routines. The normal SYM GETKEY routine goes out and waits for a key to be pressed, scanning the display while it is waiting. The equivalent KIM routine, and GETKEY2 above, go out and look once at the keyboard without scanning the display and then come back right away to the calling routine.

The normal SYM SCAND routine simply scans the display. To be compatible with the KIM system it must be capable of translation of bytes left in ØØFB,ØØFA, and ØØF9 and then displaying them. I have modified the SCANVEC to return from routine 'OUTBYT' without scanning the display during the move of the data into the display buffer because this only causes a ghost of each digit to appear on all 6 digits of the display during each scan. This way I can take advantage of the monitor data move routines and the scan the display as often as desired later by calling the normal scan routine.

ADDITION

A) LOAD the KIM Program

B) Modify the following loations: (Note - this version is not relocatable)

```
Ø2ØØ  2Ø 7Ø Ø2  START   JSR   SCAND2
Ø2Ø3  2Ø 9Ø Ø2          JSR   GETKEY2
Ø2Ø6  2Ø 75 82          JSR   ASCNIB
Ø2Ø9  BØ F5             BCS   START     ;Non-ASCII?
Ø2ØB  C9 ØA             CMP   #$ØA      ;"A" key = add
Ø2ØD  FØ Ø9             BEQ   Ø218
Ø2ØF  C9 ØB             CMP   #$ØB      ;"B" key = clear
Ø211  FØ 19             BEQ   DOGO
Ø213  BØ EB             BNE   START
Ø215  4C 39 Ø2          JMP   NUM

Ø26Ø  2Ø 86 8B  BEGIN   JSR   ACCESS
Ø263  A9 2B             LDA   #$2B      ;Modify SCANVEC
Ø265  8D 7Ø A6          STA   A67Ø
Ø268  4C ØØ Ø2          JMP   START

Ø27Ø  A5 FB     SCAND2  LDA   ØØFB
Ø272  2Ø FA 82          JSR   OUTBYT
Ø275  A5 FA             LDA   ØØFA
Ø277  2Ø FA 82          JSR   OUTBYT
Ø27A  A5 F9             LDA   ØØF9
Ø27C  2Ø FA 82          JSR   OUTBYT
Ø27F  2Ø Ø6 89          JSR   SCAND
Ø282  2Ø Ø6 89          JSR   SCAND
Ø285  2Ø Ø6 89          JSR   SCAND
Ø288  2Ø Ø6 89          JSR   SCAND
Ø28B  2Ø Ø6 89          JSR   SCAND
Ø28E  6Ø                RTS

Ø29Ø  2Ø 23 89  GETKEY2 JSR   KEYQ
Ø293  2Ø 2C 89          JSR   LRNKEY
Ø295  C5 6Ø             CMP   PREV
Ø297  FØ Ø3             BEQ   LDFF
Ø299  85 6Ø             STA   PREV
Ø29B  6Ø                RTS
Ø29C  A9 FF     LDFF    LDA   #$FF
Ø29E  6Ø                RTS
```

C) Run the program, starts at Ø26Ø

Notes: "A" key means add to total.
"B" key means clear display or clear incorrect entry.
The sequence B, Ø, A will clear the total to zero.

ASTEROID

A) LOAD the KIM program.

B) Modify the following locations: (Note - Beeper will run continuously. Simulated space ship motor.)

```
0223   EA EA EA          NOP     NOP NOP
0226   20 20 03          JSR     GETKEY2

022D   C9 01             CMPIM   #$01        ;Go left?

0248   A9 09      DISP   LDAIM   #$09        ;Set up for output
024A   20 A5 89          JSR     CONFIG
024D   A9 00              LDAIM   #00        ;INIT digit
024F   8D 02 A4          STA     A402

027A   8D 00 A4   DIGT    STA     A400        ;Light digit

027F   8D 1E A4           STA     A41E        ;STORE in timer (1/64)
0282   2C 1F A4   DELA    BIT     A41F        ;Time up?
0285   10 FB              BPL     DELA

0289   8D 00 A4           STA     A400        ;PORT A data
028C   EE 02 A4           INC     A402        ;SHIFT to next digit
028F   EA EA EA           NOP     NOP NOP     ;

02C8   4C 30 03   FIN     JMP     SCAND2

0310   20 86 8B   BEGIN   JSR     ACCESS      ;Unprotect sys RAM
0313   4C 00 02          JMP     0200

0320   20 88 81   GETKEY2 JSR     SAVER       ;SAVE registers
0323   20 23 89           JSR     KEYQ        ;READ port
0326   20 2C 89           JSR     LRNKEY      ;What key is it
0329   20 75 82           JSR     ASCN1B      ;CONVERT ASCII to hex
032C   4C B8 81           JMP     RESXAF      ;RESTORE registers

0330   A5 FB      SCAND2 LDA     00FB        ;SEND two hex digits
0332   20 FA 82           JSR     OUTBYT      ;  into display buffer
0335   A5 FA              LDA     00FA        ;Two more
0337   20 FA 82           JSR     OUTBYT      ;
033A   A5 F9              LDA     00F9        ;Last two
033C   20 FA 82           JSR     OUTBYT      ;
033F   20 06 89'  SCD     JSR     SCAND       ;SCAN display
0342   4C 3F 03           JMP     SCD         ;Continuously
```

C) Start program at 0310.

Use Key 3 to Go right, Key 1 (not Key 0) to Go left.

BAGELS

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2Ø2   2Ø 23 89            JSR     KEYQ

Ø261   2Ø ØØ Ø3  RESUME    JSR     GETKEY2

Ø271   B9 29 8C            LDA     TABLE,Y

Ø2A6   B9 29 8C            LDA     TABLE,Y

Ø2CØ   B9 29 8C            LDA     TABLE,Y

Ø2CE   A9 Ø9               LDAIM   #Ø9         ;SET port to output
Ø2DØ   2Ø A5 89            JSR     CONFIG
Ø2D3   AØ Ø5               LDY     #Ø5
Ø2D5   A2 Ø5               LDX     #Ø5

Ø2D9   8D ØØ A4            STA     SAD
Ø2DC   8C Ø2 A4            STY     SBD

Ø2E3   EA                  NOP

Ø2E8   2Ø 23 89            JSR     KEYQ

Ø2FØ   2Ø 86 8B  BEGIN     JSR     ACCESS
Ø2F3   4C ØØ Ø2            JMP     Ø2ØØ

Ø3ØØ   2Ø 88 81  GETKEY2   JRS     SAVER
Ø3Ø3   2Ø 23 89            JSR     KEYQ
Ø3Ø6   2Ø 2C 89            JSR     LRNKEY
Ø3Ø9   38                  SEC
Ø3ØA   E9 37               SBC     #$37        ;CONVERT ASCII to A-F
Ø3ØC   4C B8 81            JMP     RESXAF
```

C) RUN the program, starts at Ø2FØ.

BANDIT

A) LOAD the KIM program.

B) Modify the following locations:

```
0296   A9 09          LDAIM   #09        ;Port to output
0298   20 A5 89       JSR     CONFIG
029B   A0 00          LDYIM   #00        ;Set up INIT digit

02A1   8C 02 A4       STY     SBD
02A4   8D 00 A4       STA     SAD

02AE   C8             INY
02AF   C0 03          CPYIM   #03        ;is Y equal 3
02B1   F0 1E          BEQ     SPACER     ;ADD one space to display

02B6   20 23 89       JSR     KEYQ       ;READ keyboard, key down?

02BF   BD 29 8C       LDA     TABLE,X

02CB   BD 29 8C       LDA     TABLE,X

02D1   C8             SPACER  INY        ;ADD space between
02D2   4C B3 02                          ;Wheels and amount

02E0   20 86 8B       BEGIN   JSR     ACCESS
02E3   4C 00 02               JMP     0200
```

C) RUN the program, start address is 02E0.

Notes: Links to SYM monitor are:

```
KEYQ   -   8923
CONFIG -   89A5
SAD    -   A400
SBD    -   A402
ACCESS -   8B86
```

The added space between the wheels and the amount makes the display more readable.

BITZ

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2Ø5   2Ø 23 89   MAIN      JSR     KEYQ        ;Key down?
Ø2Ø8   2Ø 86 Ø2               JSR     GETKEY2     ;GET key input

Ø21E   BD 29 8C               LDA     TABLE,X

Ø22A   BD 29 8C               LDA     TABLE,X

Ø23B   AD 1E A4               LDA     TIMER       ;GET random value

Ø25F   A9 Ø9      LIGHT       LDAIM   #Ø9         ;Port to output
Ø261   2Ø A5 89               JSR     CONFIG
Ø264   AØ ØØ                  LDY     #ØØ

Ø26A   8D ØØ A4               STA     SAD
Ø26D   8C Ø2 A4               STY     SBD

Ø274   C8 EA                  INY     NOP

Ø28Ø   2Ø 86 8B   BEGIN       JSR     ACCESS
Ø283   4C ØØ Ø2               JMP     Ø2ØØ

Ø286   2Ø 88 81   GETKEY2     JSR     SAVER
Ø289   2Ø 23 89               JSR     KEYQ
Ø28C   2Ø 2C 89               JSR     LRNKEY
Ø28F   C9 2E                  CMP     #2E         ;All legal keys 0-9, A-F
Ø291   9Ø Ø6                  BCC     NOTKEY      ;RETURN ASCII 30-46
Ø293   2Ø 75 82               JSR     ASCNIB
Ø296   4C B8 81               JMP     RESXAF
Ø299   A9 15      NOTKEY      LDAIM   #$15        ;KIM says if 15 in ACC,
Ø29B   4C B8 81               JMP     RESXAF      ; then no key down
```

C) RUN the program - start address is Ø28Ø.

BLACKJACK

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø212   AD IE A4          LDA   TIMER

Ø332   AØ Ø5             LDY   #Ø5

Ø336   A9 Ø9             LDA   #Ø9        ;Setup for output
Ø338   2Ø A5 89          JSR   CONFIG

Ø33D   8D ØØ A4          STA   SAD

Ø34Ø   8C Ø2 A4          STY   SBD

Ø347   88 EA             DEY   NOP

Ø34C   2Ø 23 89          JSR   KEYQ
Ø34F   2Ø EC Ø3          JSR   GETKEY2

Ø3AC   B9 29 8C          LDA   TABLE,Y

Ø3B5   B9 29 8C          LDA   TABLE,Y

Ø3EC   2Ø 2C 89          JSR   LRNKEY
Ø3EF   C9 2E             CMP   #$2E       ;Key Ø-F?
Ø3F1   BØ Ø3             BCS   AKEY
Ø3F3   A9 15             LDA   #$15       ;KIM no-key
Ø3F5   6Ø                RTS              ;RETURN
Ø3F6   2Ø 75 82   AKEY   JSR   ASCNIB     ;Convert ASCII to hex
Ø3F9   6Ø                RTS              ;RETURN

Ø3FA   2Ø 86 8B   BEGIN  JSR   ACCESS
Ø3FD   4C ØØ Ø2          JMP   Ø2ØØ
```

C) RUN the program, start at location Ø3FA.

BLACK MATCH

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØC   2Ø  B5  Ø2            JSR     SCAND3
Ø2ØF   2Ø  86  Ø2            JSR     GETKEY2

Ø214   BØ  F2                BCS     PLAY

Ø23Ø   8D  1F  A4            STA     CLOCK
Ø233   2Ø  92  Ø2            JSR     SCAND2
Ø236   2C  1F  A4            BIT     CLOCK

Ø24E   AE  Ø4  AØ            LDA     AØØ4        ;Random timer
Ø275   2Ø  92  Ø2

Ø280   2Ø  86  8B            (JSR ACCESS)
Ø283   4C  ØØ  Ø2            JMP     START

Ø286   D8          GETKEY2   CLD
Ø287   2Ø  23  89            JSR     KEYQ
Ø28A   2Ø  2C  89            JSR     LRNKEY
Ø28D   2Ø  75  82            JSR     ASCNIB
Ø29Ø   F8                    SED
Ø291   6Ø                    RTS
Ø292   D8          SCAND2    CLD
Ø293   A9  2B                LDA     #2B         ;Change
Ø295   8D  7Ø  A6            STA     A670        ;SCANVEC
Ø298   A5  FB                LDA     ØØFB
Ø29A   2Ø  FA  82            JSR     OUTBYT
Ø29D   A5  FA                LDA     ØØFA
Ø29F   2Ø  FA  82            JSR     OUTBYT
Ø2A2   A5  F9                LDA     ØØF9
Ø2A4   2Ø  FA  82            JSR     OUTBYT
Ø2A7   2Ø  Ø6  89            JSR     SCAND
Ø2AA   2Ø  Ø6  89            JSR     SCAND
Ø2AD   2Ø  Ø6  89            JSR     SCAND
Ø2BØ   2Ø  Ø6  89            JSR     SCAND
Ø2B3   F8                    SED
Ø2B4   6Ø                    RTS
Ø2B5   A5  F9        SCAND3  LDA     ØØF9
Ø2B7   C9  21                CMP     #$21
Ø2B9   FØ  Ø3                BEQ     INIT
Ø2BA   4C  92  Ø2            JMP     SCAND2
Ø2BD   A9  ØØ                LDA     #$ØØ
Ø2BF   85  FA                STA     ØØFA
Ø2C1   4C  92  Ø2            JMP     SCAND2
```

C) RUN the program, start at Ø28Ø.

CARD DEALER

A) LOAD the KIM program

B) Modify the following locations:

```
0018   4C 06 02            JMP    RESTART
001B   AD 1E A4            LDA    A41E

0020   AD 04 A0            LDA    A004

0062   20 88 81  DISP      JSR    SAVER
0065   A2 00              LDX    #$00
0067   B5 8C      MOVER    LDA    008C,X     ;GET table value
0069   9D 40 A6           STA    A640,X     ;STORE in disbuf
006C   E8                 INX
006D   E0 06              CPX    #$06       ;6 digits yet?
006F   90 F6              BCC    MOVER      ;No, do again
0071   20 06 89           JSR    SCAND      ;Scand Display
0074   20 23 89           JSR    KEYQ       ;Key down?
0077   4C B8 81           JMP    RESXAF     ;RESTORE registers
007A   EA                 NOP

0200   20 86 8B  BEGIN     JSR    ACCESS
0203   4C 00 00           JMP    INIT
0206   20 AF 88  RESTART   JSR    GETKEY
0209   20 72 89           JSR    BEEP
020C   20 72 89           JSR    BEEP
020F   20 72 89           JSR    BEEP
0212   20 72 89           JSR    BEEP
0215   4C 00 00           JMP    START
```

C) RUN the program, start at 0200.

Note: After 52 cards are dealt, beeper will ripple indicating reshuffle; first card after beep is first card of a new deck. Use Key 2 to select cards for best results.

CHESS CLOCK

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØØ   AA A9 ØØ          TAX ,   LDA #$ØØ
Ø2Ø3   9D 7A ØØ          STA     Ø07A,X
Ø2Ø6   CA                DEX

Ø2Ø9   2Ø AF Ø2  DISP    JSR     SCAND2
Ø2ØC   2Ø CD Ø2          JSR     GETKEY2

Ø267   8D 1F A4          STA     A41F        ;Timer 1024
Ø26A   2Ø AF Ø2          JSR     SCAND2
Ø26D   2Ø CD Ø2          JSR     GETKEY2

Ø275   2C Ø5 A4          VIT     A405        ;Time done?

Ø280   8D 1E A4          STA     A41E        ;Timer 64
Ø283   2C Ø5 A4          BIT     A4Ø5

Ø2A1   20 86 8B  BEGIN   JSR     ACCESS
Ø2A4   A9 2B             LDA     #$2B        ;Modify SCANVEC
Ø2A6   8D 7Ø A6          STA     A67Ø
Ø2A9   A9 F8             LDA     #$F8        ;Arrange zeros in ØØ-F8 only
Ø2AB   4C ØØ Ø2
Ø2AE   EA
Ø2AF   D8        SCAND2  CLD
Ø2BØ   A5 FB 2Ø  FA 82   LDA     ØØFB/JSR OUTBYT
Ø2B5   A5 FA 2Ø  FA 82   LDA     ØØFA/JSR OUTBYT
Ø2BA   A5 F9 2Ø  FA 82   LDA     ØØF9/JSR OUTBYT
Ø2BF   2Ø Ø6 89  2Ø Ø6   JSR     SCAND (X4)
  89   2Ø Ø6    89 2Ø
  Ø6   89
Ø2CB   F8 6Ø             SED/RTS

Ø2CD   D8        GETKEY2 CLD
Ø2CE   2Ø 88 81          JSR     SAVER
Ø2D1   2Ø 23 89          JSR     KEYQ
Ø2D4   2Ø 2C 89          JSR     LRNKEY
Ø2D7   2Ø 75 82          JSR     ASCNIB
Ø2DA   F8                SED
Ø2DB   4C B8 81          JMP     RESXAF
```

C) RUN the program, start at Ø2A1.

CODE TEST

A) LOAD the KIM program.

Note: This routine as modified does not need an external speaker. Instead the beeper is used.

B) Modify the following locations:

```
Ø256  2Ø 23 89          JSR   KEYQ

Ø286  8D 1F A4          STA   A41F      ;Timer 1024
Ø289  2Ø 2Ø Ø1          JSR   DISBEEP   ;Light parts
Ø28C  2Ø 2Ø Ø1          JSR   DISBEEP   ;Of digit 6
Ø28F  2Ø 2Ø Ø1          JSR   DISBEEP   ;And beep
Ø292  2Ø 2Ø Ø1          JSR   DISBEEP   ;Onboard SPKR
Ø295  EA                NOP
Ø296  2C Ø5 A4          BIT   A4Ø5      ;Time up?

Ø2A4  8D 1F A4          STA   A41F      ;Start timer
Ø2A7  2C Ø5 A4          BIT   A4Ø5      ;Time up?

Ø2B1  A9 Ø9             LDA   #$Ø9      ;Set for
Ø2B3  2Ø A5 89          JSR   CONFIG    ;Output

Ø2B8  A2 ØØ             LDX   #ØØ      ;INIT Digit

Ø2BF  2Ø ØØ Ø1          JSR   DISPL

Ø2C7  2Ø 23 89          JSR   KEYQ      ;Key down?

Ø1ØØ  8D ØØ A4          STA   PAD
Ø1Ø3  8E Ø2 A4          STX   PBD
Ø1Ø6  E8                INX
Ø1Ø7  E6 AA       DELAYA INC   AA

Ø1Ø9  DØ FC             BNE   DELAYA
Ø1ØB  6Ø

Ø11Ø  2Ø 86 8B 4C ØØ Ø2 BEGIN JSR ACCESS/JMP Ø2ØØ

Ø12Ø  A9 Ø9       DISBEEP LDA   #$Ø9       ;Set up for
Ø122  2Ø A5 89            JSR   CONFIG     ; output
Ø125  A9 Ø5 8D Ø2 A4      LDA   #$Ø5/STA A402 ;SELECT digit 6
Ø12A  A9 8Ø EØ Ø1 FØ Ø2   LDA   #$8Ø/CPYIM #Ø1/BEQ OUTS
Ø13Ø  A9 Ø8               LDA   #$Ø8
Ø132  8D ØØ A4    OUTS    STA   A4ØØ
Ø135  E6 AA DO FC DELAYB  INC   AA/BNE DELAYB
Ø139  2Ø 72 89            JSR   BEEP
Ø13C  6Ø                  RTS
```

C) RUN the program, start at Ø11Ø.

CRAPS

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2Ø1   2Ø 86 8B          JSR    ACCESS
Ø2Ø4   2Ø D3 Ø2          JSR    GETKEY2

Ø21A   AD 1E A4          LDA    TIMER      ;Random value

Ø23Ø   BD 29 8C          LDA    TABLE,X    ;Segment

Ø28C   A9 Ø9      NOINC  LDA    #$Ø9       ;Port to output
Ø28E   2Ø A5 89          JSR    CONFIG     ;
Ø291   AØ Ø5             LDY    #$Ø5       ;

Ø297   8D ØØ A4          STA    SAD
Ø29A   8C Ø2 A4          STY    SBD

Ø2A1   88 EA             DEY    NOP

Ø2B9   BD 29 8C          LDA    TABLE,X

Ø2C2   BD 29 8C          LDA    TABLE,X

Ø2D3   2Ø 88 81          JSR    SAVER      ;SAVE registers
Ø2D6   2Ø 23 89          JSR    KEYQ       ;READ input port
Ø2D9   2Ø 2C 89          JSR    LRNKEY     ;Translate to ASCII
Ø2DC   C9 2F             CMP    #$2F       ;ASCII 30-46?
Ø2DE   BØ Ø3             BCS    AKEY       ;Yes
Ø2EØ   A9 15             LDA    #$15       ;KIM nokey indicator
Ø2E2   4C B8 81          JMP    RESXAF     ;RESTORE registers
Ø2E5   2Ø 75 82          JSR    ASCNIB     ;CONVERT ASCII to hex
Ø2E8   4C B8 81          JMP    RESXAF     :RESTORE registers
```

C) RUN the program, start at location Ø215.

Notes:

1) Program initializes to a $10 pot at the start. Reset and restart at Ø215 any time desired.

2) If GO key is used to roll the dice, impact on total display is minimized.

DUEL

A) LOAD the KIM program, deleting section from Ø271 - Ø2A5

B) Modify the following locations:

```
Ø2Ø6   AD Ø8 AØ  RAND     LDA    AØØ8          ;GET radom Number

Ø2Ø6   A9 2Ø             LDA    #$2Ø          ;ASCII blank

Ø216   2C Ø5 A4          BIT    A4Ø5          ;Time up ?
Ø219   DØ ØD             BNE    MORE          ;

Ø21D   8D 1F A4          STA    A41F          ;START timer

Ø224   A9 2D             LDA    #$2D          ;Center segment only for DIG.3,4

Ø229   2Ø 23 89          JSR    KEYQ          ;
Ø22C   2Ø AØ Ø2          JSR    GETKEY2       ;

Ø237   C9 Ø4             CMP    #$Ø4          ;change left player to 4 key

Ø264   2Ø 23 89          JSR    KEYQ          ;

Ø271   D8          LITE   CLD                  ;Clear DEC. mode
Ø272   A9 2B             LDA    #$2B          ;  DELETE scan during
Ø274   8D 7Ø A6          STA    A67Ø          ;  Data transfer
Ø277   A5 FB             LDA    ØØFB          ;
Ø279   2Ø FA 82          JSR    OUTBYT        ;
Ø27C   A5 FA             LDA    ØØFA          ;
Ø27E   2Ø 47 8A          JSR    OUTCHR        ;
Ø281   2Ø 47 8A          JSR    OUTCHR        ;
Ø284   A5 F9             LDA    ØØF9          ;
Ø286   2Ø FA 82          JSR    OUTBYT        ;
Ø289   AD 97 Ø2          LDA    Ø297          ;
Ø28C   8D AD Ø2          STA    Ø2AD          ;
Ø28F   2Ø Ø6 89    SC2    JSR    SCAND         ;
Ø292   CE AD Ø2          DEC    Ø2AD          ;
Ø295   DØ F8 ØA          BNE    SC2/.BYTEØA   ;Display brightness
Ø298   F8                SED                  ;
Ø299   6Ø                RTS                  ;
Ø29A   2Ø 86 8B    BEGIN  JSR    ACCESS        ;
Ø29D   4C ØØ Ø2          JMP    Ø2ØØ          ;
Ø2AØ   2Ø 88 81    GETKEY2 JSR   SAVER         ;
Ø2A3   2Ø 2C 89          JSR    LRNKEY        ;
Ø2A6   2Ø 75 82          JSR    ASCNIB        ;
Ø2A9   4C B8 81          JMP    RESXAF        ;
Ø2AD   ØØ                .BYTE                ;Workspace
```

C) RUN program, starts at Ø29A.

Note: Left player uses key 4, not zero.

Center two digits display hyphens after timeout.

If either player hits key before hyphens lit, he loses a point.

FARMER BROWN

A) LOAD the KIM program

B) Modify the following locations:

```
Ø22Ø   4C E6 Ø2          JMP   DONE      :out of corn, show ERMSG
Ø223   AD IE A4  MORE    LDA   TIMER     ;

Ø255   2Ø 23 89          JSR   KEYQ
Ø258   2Ø DØ Ø2          JSR   GETKEY2

Ø287   A9 Ø9             LDA   #$Ø9      ;Port to
Ø289   2Ø A5 89          JSR   CONFIG    ; output
Ø28C   AØ Ø5             LDY   #$Ø5

Ø292   8D ØØ A4          STA   SAD
Ø295   8C Ø2 A4          STY   SBD

Ø29C   88 EA CA          DEY   NOP DEX

Ø2DØ   2Ø 88 81  GETKEY2 JSR   SAVER
Ø2D3   2Ø 2C 89          JSR   LRNKEY
Ø2D6   2Ø 75 82          JSR   ASCNIB
Ø2D9   4C B8 81          JMP   RESXAF

Ø2EØ   2Ø 86 8B  BEGIN   JSR   ACCESS
Ø2E3   4C ØØ Ø2          JMP   START
Ø2E6   A9 ØØ             LDA   #$ØØ
Ø2E8   38                SEC
Ø2E9   4C 71 81          JMP   ERMSG
```

C) RUN the program, start at Ø2EØ.

Note: When you run out of corn, display will show "Er ØØ." and will return to the monitor program.

HI-LO

A) Enter the KIM Program

B) Modify the following locations:

```
Ø2ØF   2Ø 23 89          JSR    KEYQ

Ø223   2Ø 7Ø Ø2  GUESS   JSR    SCAND2
Ø226   2Ø 9Ø Ø2          JSR    GETKEY2
Ø229   C9 17             CMP    #$17      ;Go key?

Ø233   C9 11             CMP    #$11      ;"A" key?

Ø26A   2Ø 86 8B  BEGIN   JSR    ACCESS
Ø26D   4C ØØ Ø2          JMP    START

Ø27Ø   A9 2B     SCAND2  LDA    #$2B      ;Modify SCANVEC
Ø272   8D 7Ø A6          STA    A67Ø
Ø275   A5 FB             LDA    ØØFB
Ø277   2Ø FA 82          JSR    OUTBYT
Ø27A   A5 FA             LDA    ØØFA
Ø27C   2Ø FA 82          JSR    OUTBYT
Ø27F   A5 F9             LDA    ØØF9
Ø281   2Ø FA 82          JSR    OUTBYT
Ø284   2Ø Ø6 89          JSR    SCAND
Ø287   2Ø Ø6 89          JSR    SCAND
Ø28A   2Ø Ø6 89          JSR    SCAND
Ø28D   6Ø                RTS

Ø29Ø   2Ø 88 81  GETKEY2 JSR    SAVER
Ø293   2Ø 23 89          JSR    KEYQ
Ø296   2Ø 2C 89          JSR    LRNKEY
Ø299   38                SEC
Ø29A   E9 3Ø             SBC    #$3Ø
Ø29C   4C B8 81          JMP    RESXAF
```

C) Run the program, starts at Ø26A.

HORSERACE

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØB   A9 Ø9      DISP      LDA     #$Ø9          ;Port to
Ø2ØD   2Ø A5 89             JSR     CONFIG        ; output

Ø212   EA EA                NOP     NOP

Ø219   2Ø Ø6 Ø3             JSR     DISOUT        ;Light display

Ø221   EA EA EA             NOP     NOP NOP

Ø29F   2Ø 16 Ø3             JSR     GETKEY2

Ø3ØØ   2Ø 86 8B   BEGIN     JSR     ACCESS
Ø3Ø3   4C ØØ Ø2             JMP     Ø2ØØ

Ø3Ø6   8D ØØ A4   DISOUT    STA     A400
Ø3Ø9   8C Ø2 A4             STY     A402
Ø3ØC   E6 4Ø      WAITR     DEC     WAIT
Ø3ØE   DØ FC                BNE     WAITR
Ø31Ø   A9 ØØ                LDA     #ØØ
Ø312   8D ØØ A4             STA     A4ØØ

Ø315   6Ø                   RTS

Ø316   2Ø 88 81   GETKEY 2  JSR     SAVER
Ø319   2Ø 23 89             JSR     KEYQ
Ø31C   2Ø 2C 89             JSR     LRNKEY
Ø31F   2Ø 75 82             JSR     ASCNIB
Ø322   4C B8 81             JMP     RESXAF
```

C) RUN the program, start at Ø3ØØ.

Notes: Whipping keys are set at locations 2FØ, 2F1, 2F2, now set so that the 4 key controls the top segment, and 2 key controls the middle and the 1 key the bottom.

KEYTRAIN

A) LOAD the KIM program.

B) Modify the following locations:

```
0000  20 23 89  START   JSR   KEYQ

0005  AD 1E A4          LDA   TIMER

001A  20 0C 02  LIGHT   JSR   SCAND2
001D  20 28 02          JSR   GETKEY2

0200  D8        BEGIN   CLD
0201  20 86 8B          JSR   ACCESS
0204  A9 2B             LDA   #$2B      ;Setup for
0206  8D 70 A6          STA   A670      ; no scan during move of data
0209  4C 00 00          JMP   0000

020C  A5 F9     SCAND2  LDA   00F9
020E  20 FA 82          JSR   OUTBYT
0211  A5 FA             LDA   00FA
0213  20 FA 82          JSR   OUTBYT
0216  A5 FB             LDA   00FB
0218  20 FA 82          JSR   OUTBYT
021B  20 06 89          JSR   SCAND
021E  20 06 89          JSR   SCAND
0221  20 06 89          JSR   SCAND
0224  20 06 89          JSR   SCAND
0227  60                RTS

0228  20 88 81  GETKEY2 JSR   SAVER
022B  20 23 89          JSR   KEYQ
022E  20 2C 89          JSR   LRNKEY
0231  20 75 82          JSR   ASCNIB
0234  4C B8 81          JMP   RESXAF
```

C) RUN the program, START at 0200.

KIM NIM

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØØ  86 8B  START      JSR    ACCESS
Ø2Ø3  2Ø 5Ø  Ø3          JSR    GETKEY2

Ø2ØA  AD 1E  A4          LDA    TIMER

Ø22Ø  2Ø 23  89  STALL   JSR    KEYQ

Ø225  AD 1E  A4          LDA    TIMER

Ø28D  A9 Ø9              LDA    #$Ø9
Ø28F  2Ø A5  89          JSR    CONFIG    ;Set for output
Ø292  AØ Ø5      LIGHT   LDY    #Ø5

Ø298  8D ØØ  A4          STA    SAD
Ø29B  8C Ø2  A4          STY    SBD

Ø2A2  88 EA              DEY    NOP

Ø2D2  CC 1E  A4          CPY    TIMER

Ø3Ø2  4C Ø3  Ø2          JMP    START

Ø332  B9 29  8C          LDA    TABLE,Y

Ø35Ø  D8         GETKEY2 CLD
Ø351  2Ø 88  81          JSR    SAVER
Ø354  2Ø 23  89          JSR    KEYQ
Ø357  2Ø 2C  89          JSR    LRNKEY
Ø35A  2Ø 75  82          JSR    ASCNIB
Ø35D  DØ Ø2              BNE    GOBACK
Ø35F  A9 13              LDA    #$13
Ø361  F8         GOBACK  SED
Ø362  4C B8  81          JMP    RESXAF
```

C) RUN the program, START at Ø2ØØ.

Note: Press "CR" at any time to start a new game (not GO key).

KIM-TAC-TOE (continued)

```
0181  A5 E9          LDA   00E9
0183  20 FA 82       JSR   OUTBYT
0186  A9 06          LDA   #$06      ;RESTORE
0188  8D 70 A6       STA   A670      ;SCANVEC
018B  20 BC 01       JSR   SCAND3    ;SCAN display once
018E  60             RTS
018F  EA             NOP
0190  85 AA    DISOUT STA   00AA
0192  A9 00          LDA   #00
0194  8D 00 A4       STA   A400      ;Turn off display
0197  A5 AA          LDA   00AA      ;LOAD new data
0199  8E 02 A4       STX   A402      ;Turn on display again
019C  8D 00 A4       STA   A400
019F  E8 60          INX   RTS       ;Increment X, Return

01A1  20 86 8B BEGIN  JSR   ACCESS
01A4  4C 00 01        JMP   0100
01A7  85 AA           STA   00AA      ;SAVE key ASCII-code
01A9  20 75 82        JSR   ASCNIB    ;connect to hex if hex
01AC  90 0B           BCC   RETN      ;If was hex, go back
01AE  A5 AA           LDA   00AA      ;If not hex, try again
01B0  38              SEC
01B1  E9 30           SBC   #$30      ;Subtract 30H
01B3  C9 22           CMP   #$22      ;Was it "REG" key?  LOAD "10"
01B5  D0 02           BNE   RETN      ;If not RETURN direct
01B7  A9 10           LDA   #$10
01B9  4C B8 81 RETN   JMP   RESXAF    :RESTORE registers
01BC  20 06 89 SCAND3 JSR   SCAND     ;SCAN the display to show odds
01BF  20 06 89        JSR   SCAND
01C2  20 06 89        JSR   SCAND
01C5  20 06 89        JSR   SCAND
01C8  20 06 89        JSR   SCAND
01CB  60              RTS
```

C) RUN the program, START AT 01A1.

Notes: After starting, press either GO or +/- to begin game.
"MEM" key is used instead of "PC" key to see odds.
"REG" key is used instead of "DA" key to return to game.
"+/-" key is used to reset for users first selection.
"Go" key is used as text specifies to reset after end of game for machine to go first.
User markers are brighter than machine markers/SYM's markers flash.
During a win, user markers flash also but at a different flash rate than the SYM's.

KIM-TAC-TOE (continued)

```
Ø181  A5 E9          LDA  ØØE9
Ø183  2Ø FA 82       JSR  OUTBYT
Ø186  A9 Ø6          LDA  #$Ø6      ;RESTORE
Ø188  8D 7Ø A6       STA  A67Ø      ;SCANVEC
Ø18B  2Ø BC Ø1       JSR  SCAND3    ;SCAN display once
Ø18E  6Ø             RTS
Ø18F  EA             NOP
Ø19Ø  85 AA    DISOUT STA  ØØAA
Ø192  A9 ØØ          LDA  #ØØ
Ø194  8D ØØ A4       STA  A4ØØ      ;Turn off display
Ø197  A5 AA          LDA  ØØAA      ;LOAD new data
Ø199  8E Ø2 A4       STX  A4Ø2      ;Turn on display again
Ø19C  8D ØØ A4       STA  A4ØØ
Ø19F  E8 6Ø          INX  RTS       ;Increment X, Return

Ø1A1  2Ø 86 8B BEGIN JSR  ACCESS
Ø1A4  4C ØØ Ø1       JMP  Ø1Ø0
Ø1A7  85 AA          STA  ØØAA      ;SAVE key ASCII-code
Ø1A9  2Ø 75 82       JSR  ASCNIB    ;connect to hex if hex
Ø1AC  9Ø ØB          BCC  RETN      ;If was hex, go back
Ø1AE  A5 AA          LDA  ØØAA      ;If not hex, try again
Ø1BØ  38             SEC
Ø1B1  E9 3Ø          SBC  #$3Ø      ;Subtract 3ØH
Ø1B3  C9 22          CMP  #$22      ;Was it "REG" key? LOAD "1Ø"
Ø1B5  DØ Ø2          BNE  RETN      ;If not RETURN direct
Ø1B7  A9 1Ø          LDA  #$1Ø
Ø1B9  4C B8 81 RETN  JMP  RESXAF    :RESTORE registers
Ø1BC  2Ø Ø6 89 SCAND3 JSR SCAND     ;SCAN the display to show odds
Ø1BF  2Ø Ø6 89       JSR  SCAND
Ø1C2  2Ø Ø6 89       JSR  SCAND
Ø1C5  2Ø Ø6 89       JSR  SCAND
Ø1C8  2Ø Ø6 89       JSR  SCAND
Ø1CB  6Ø             RTS
```

C) RUN the program, START AT Ø1A1.

Notes: After starting, press either GO or +/- to begin game.
"MEM" key is used instead of "PC" key to see odds.
"REG" key is used instead of "DA" key to return to game.
"+/-" key is used to reset for users first selection.
"Go" key is used as text specifies to reset after end of game for machine to go first.
User markers are brighter than machine markers/SYM's markers flash.
During a win, user markers flash also but at a different flash rate than the SYM's.

KIM NIM

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØØ  86 8B  START     JSR    ACCESS
Ø2Ø3  2Ø 5Ø  Ø3        JSR    GETKEY2

Ø2ØA  AD 1E  A4        LDA    TIMER

Ø22Ø  2Ø 23  89  STALL  JSR    KEYQ

Ø225  AD 1E  A4        LDA    TIMER

Ø28D  A9 Ø9            LDA    #$Ø9
Ø28F  2Ø A5  89        JSR    CONFIG    ;Set for output
Ø292  AØ Ø5     LIGHT  LDY    #Ø5

Ø298  8D ØØ  A4        STA    SAD
Ø29B  8C Ø2  A4        STY    SBD

Ø2A2  88 EA            DEY    NOP

Ø2D2  CC 1E  A4        CPY    TIMER

Ø3Ø2  4C Ø3  Ø2        JMP    START

Ø332  B9 29  8C        LDA    TABLE,Y

Ø35Ø  D8        GETKEY2 CLD
Ø351  2Ø 88  81        JSR    SAVER
Ø354  2Ø 23  89        JSR    KEYQ
Ø357  2Ø 2C  89        JSR    LRNKEY
Ø35A  2Ø 75  82        JSR    ASCNIB
Ø35D  DØ Ø2            BNE    GOBACK
Ø35F  A9 13            LDA    #$13
Ø361  F8        GOBACK SED
Ø362  4C B8  81        JMP    RESXAF
```

C) RUN the program, START at Ø2ØØ.

Note: Press "CR" at any time to start a new game (not GO key).

MULTI-MAZE

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØØ  AD 1E A4  START   LDA   TIMER
Ø2Ø3  85 DØ             STA   ØØDØ
Ø2Ø5  EA EA             NOP   NOP

Ø267  A9 Ø9     MUG     LDA   #$Ø9        ;PORT to output
Ø269  2Ø A5 89          JSR   CONFIG
Ø26C  AØ ØØ             LDY   #$ØØ

Ø272  8D ØØ A4          STA   SAD
Ø275  8C Ø2 A4          STY   SBD

Ø27D  EA                NOP

Ø282  2Ø 23 89          JSR   KEYQ
Ø285  2Ø 3Ø Ø3          JSR   GETKEY2

Ø2CE  47 39 31  TAB2    'G', '9',  '1', '6', '4'
Ø2D1  36 34

Ø33Ø  2Ø 88 81  GETKEY2 JSR   SAVER
Ø333  2Ø 23 89          JSR   KEYQ
Ø336  2Ø 2C 89          JSR   LRNKEY
Ø339  4C B8 81          JMP   RESXAF

Ø34Ø  2Ø 86 8B  BEGIN   JSR   ACCESS
Ø343  4C ØØ Ø2          JMP   Ø2ØØ
```

C) RUN the program, START at Ø34Ø.

PING PONG

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2ØØ   2Ø 23 89   START     JSR     KEYQ
Ø2Ø3   2Ø 59 Ø3             JSR     GETKEY2
Ø2Ø6   C9 Ø1                CMP     #$Ø1        ;"1" key?

Ø214   EA EA      NOGO      NOP     NOP
Ø216   9Ø 22                BCC     NOKEY       ;Carry set for nonhex

Ø23A   2Ø 23 89   NOKEY     JSR     KEYQ

Ø266   A9 Ø9      FREEZE    LDA     #$Ø9        ;Port to output
Ø268   2Ø A5 89             JSR     CONFIG
Ø26B   AØ Ø5                LDY     #$Ø5

Ø27E   BD 29 8C   HOOP      LDA     TABLE,X

Ø2A4   8D ØØ A4   SHOW      STA     SAD
Ø2A7   8C Ø2 A4             STY     SBD

Ø2AE   88 EA                INY     NOP

Ø2DE   AD 1E A4             LDA     TIMER

Ø359   2Ø 88 81   GETKEY2   JSR     SAVER
Ø35C   2Ø 2C 89             JSR     LRNKEY
Ø35F   2Ø 75 82             JSR     ASCNIB
Ø362   4C B8 81             JMP     RESXAF

Ø365   2Ø 86 8B   BEGIN     JSR     ACCESS
Ø368   4C ØØ Ø2             JMP     Ø2ØØ
```

C) RUN the program, START at Ø365.

Notes: "1" key is used to reset to a new game.
All other instructions are valid as written.

QUICK

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø3Ø9   EA EA EA          NOP     NOP NOP
Ø3ØC   EA EA             NOP     NOP

Ø326   2Ø 2Ø Ø2          JSR     SCAND2

Ø32B   2Ø Ø6 89  STAND   JSR     SCAND2
Ø32E   2Ø 35 Ø2          JSR     GETKEY2

Ø22Ø   D8        SCAND2  CLD             ;CLEAR decimal
Ø221   A5 FB             LDA     ØØFB
Ø223   2Ø FA 82          JSR     OUTBYT
Ø226   A5 FA             LDA     ØØFA
Ø228   2Ø FA 82          JSR     OUTBYT
Ø22B   A5 F9             LDA     ØØF9
Ø22D   2Ø FA 82          JSR     OUTBYT
Ø23Ø   2Ø Ø6 89          JSR     SCAND
Ø233   F8                SED             ;Set decimal
Ø234   60                RTS

Ø235   D8        GETKEY2 CLD
Ø236   2Ø 88 81          JSR     SAVER
Ø239   2Ø 23 89          JSR     KEYQ
Ø23C   2Ø 2C 89          JSR     LRNKEY
Ø23F   C9 47             CMP     #$47
Ø241   FØ Ø7             BEQ     GKEY
Ø243   F8                SED
Ø244   2Ø 75 82          JSR     ASCNIB
Ø247   4C B8 81          JMP     RESXAF
Ø24A   A9 13     GKEY    LDA     #$13
Ø24C   F8                SED
Ø24D   4C B8 81          JMP     RESXAF

Ø21Ø   2Ø 86 8B  BEGIN   JSR     SAVER
Ø213   A9 2B             LDA     #$2B    ;Modify SCANVEC
Ø215   8D 7Ø A6          STA     A67Ø
Ø217   4C ØØ Ø3          JMP     START
```

C) RUN the program, START at Ø21Ø.

REVERSE

A) LOAD the KIM program.

B) Modify the following locations:

```
Ø2Ø2   2Ø 23 89          JSR    KEYQ

Ø239   B9 33 8C          LDA    TABLE+1Ø,Y

Ø262   A9 Ø9             LDA    #$Ø9
Ø264   2Ø A5 89          JSR    CONFIG
Ø267   AØ ØØ             LDY    #ØØ

Ø26D   8D ØØ A4          STA    SAD
Ø27Ø   8C Ø2 A4          STY    SBD

Ø277   EA                NOP

Ø27C   2Ø 23 89          JSR    KEYQ
Ø27F   2Ø BØ Ø2          JSR    GETKEY2
Ø282   EA EA             NOP    NOP
Ø284   BØ C5             BCS    SLINK        ;Carry set if not hex char.

Ø2A6   77 7C 39 WINNER   .BYTE  $77,$7C,$39,$5E,$79,$71
Ø2A9   5E 79 71

Ø2BØ   2Ø 88 81 GETKEY2  JSR    SAVER
Ø2B3   2Ø 2C 89          JSR    LRNKEY
Ø286   2Ø 75 82          JSR    ASCNIB
Ø2B9   4C B8 81          JMP    RESXAF

Ø2CØ   2Ø 86 8B BEGIN    JSR    ACCESS
Ø2C3   4C ØØ Ø2          JMP    START
```

C) RUN the program, START at Ø2CØ.

Note: "CR" key serves as reset for start of new game.

TEASER

A) LOAD the KIM program but do not enter Ø2DD-Ø2FA.

B) Modify the following locations:

```
Ø21B  2Ø 23 89          JSR   KEYQ

Ø22Ø  2Ø 23 89          JSR   KEYQ

Ø225  2Ø Ø1 Ø3          JSR   GETKEY2
Ø228  C9 17             CMP   #$17        ;Go key?

Ø235  2Ø FA Ø2          JSR   SEG         ;Convert to segments

Ø25A  2Ø FA Ø2          JSR   SEG

Ø261  2Ø F6 Ø2          JSR   LEFT

Ø26D  2Ø 1Ø Ø3          JSR   SCN

Ø2BE  2Ø 23 89          JSR   KEYQ
Ø2C1  2Ø Ø1 Ø3          JSR   GETKEY2
Ø2C4  C9 17             CMP   #$17

Ø2DD  2Ø 88 81  DISP    JSR   SAVER       ;Display routine
Ø2EØ  AØ ØØ             LDY   #$ØØ
Ø2E2  A2 Ø0             LDX   #$ØØ
Ø2E4  B5 DØ     DLOAD   LDA   ØØDØ,X      ;Move contents of
Ø2E6  99 4Ø A6          STA   A64Ø,Y      ; DØ,D2,D4,D6,D8,DA
Ø2E9  E8 E8 C8          INX   INX INY     ; into disbuff
Ø2EC  CØ Ø6             CPY   #$Ø6        ;6 digits moved?
Ø2EE  9Ø F4             BCC   DLOAD
Ø2FØ  2Ø Ø6 89          JSR   SCAND
Ø2F3  4C C4 81          JMP   RESALL

Ø2F6  4A 4A 4A 4A LEFT  (LSR A) *4        ;Left shift routine
Ø2FA  29 ØF             AND   #$ØF
Ø2FC  A8                TAY
Ø2FD  B9 29 8C          LDA   TABLE,Y
Ø3ØØ  60                RTS
Ø3Ø1  2Ø 88 81  GETKEY2 JSR   SAVER
Ø3Ø4  2Ø 2C 89          JSR   LRNKEY
Ø3Ø7  38 E9 3Ø          SEC,  SBC #$3Ø
Ø3ØA  4C B8 81          JMP   RESXAF

Ø31Ø  2Ø 88 81  SCN     JSR   SAVER
Ø313  2Ø Ø6 89          JSR   SCAND
Ø316  4C C4 81          JMP   RESALL

Ø32Ø  2Ø 86 8B  BEGIN   JSR   ACCESS
Ø323  4C ØØ Ø2          JMP   BEGN
```

C) RUN the program, START at Ø32Ø.

WUMPUS

A) LOAD the KIM program

B) Modify the following locations:

```
Ø35B   B9 29 8C          LDA   TABLE,Y

Ø364   B9 29 8C          LDA   TABLE,Y

Ø375   EA EA             NOP   NOP
Ø377   BØ 48             BCS   ROOM      ;Note - ROOM is Ø3C1, TYPO
                                         ;  ERROR duplicated room at Ø36C
Ø22E   8D IF A4          STA   TIMESET   ;1024 cycle timer

Ø234   2C IF A4          BIT   A41F      ;Time up?

Ø23E   A9 Ø9             LDA   #$Ø9
Ø24Ø   2Ø A5 89          JSR   CONFIG

Ø245   A2 ØØ             LDX   #$ØØ

Ø24A   EA EA             NOP   NOP
Ø24C   2Ø B6 Ø1          JSR   DISOUT

Ø254   2Ø 23 89 6Ø       JSR   KEYQ/RTS
Ø258   EA EA EA  DEBO    NOP   NOP NOP

Ø26A   2Ø C1 Ø1          JSR   GETKEY2
Ø26D   EA EA             NOP   NOP
Ø26F   EA EA             NOP   NOP

Ø2A5   AD IE A4          LDA   TIMER

Ø2EØ   B9 29 BC          LDA   TABLE,Y

Ø1BØ   2Ø 86 8B  BEGIN   JSR   ACCESS
Ø1B3   4C Ø5 Ø3          JMP   Ø3Ø5

Ø1B6   8D ØØ A4  DISOUT  STA   A4ØØ
Ø1B9   8C Ø2 A4          STY   A4Ø2
Ø1BC   E6 46     WAITR   INC   WAIT
Ø1BE   DØ FC             BNE   WAITR
Ø1CØ   6Ø                RTS

ØC1    2Ø 88 81  GETKEY2 JSR   SAVER
Ø1C4   2Ø 23 89          JSR   KEYQ
Ø1C7   2Ø 2C 89          JSR   LRNKEY
Ø1CA   2Ø 75 82          JSR   ASCNIB
Ø1CD   4C B8 81          JMP   RESXAF
```

C) RUN the program, START at Ø1BØ

Note: "GO" key is used to pitch can of gas.

Rate at location Ø229 should be "Ø3" for a decent display speed.