STARLASER

FEATURED GAME

In this Hi-Res *action game, the object is to shoot down as many Kloryon ships as you can before you run out of fuel.*

Enemy Kloryon starbases are invading your sector of the galaxy. Your mission, as captain of an allied starship, is to destroy as many starbases as possible while your fuel supply lasts. To neutralize your laser fire, the Kloryons have generated an asteroid belt between your starship and their starbases. If the laser beam strikes an asteroid, its power is neutralized. Good luck, Captain!

PLAYING STARLASER

A warning siren sounds as each Kloryon starbase appears on the right side of the screen. The moving asteroid belt appears in the center of the screen. Your starship is on the left. (See **Figure 1**.) To accelerate your starship down the screen (or to decelerate as you move up), press the Right-Arrow key; to accelerate up the screen (or to decelerate as you move down), press the Left-Arrow key. To fire your lasers, press the Space bar.

Starlaser has two auxiliary commands: S toggles the sound effects on or off (via the &STOP and &NORMAL commands of DUO, the dual-pitched sound routine that is described in the article "Arcade Sound Editor," also in this issue). Escape pauses the program.

Your fuel capacity is 1,000 units. The faster your starship travels, the faster you use up fuel. Firing a laser also uses up fuel, 10 units per blast. When your fuel is gone, the game is over, and your final score is displayed.

The score is based upon the number of Kloryon starbases that you destroy. You get 50 points for each. You lose points for firing into an asteroid.

At the end of each game, you are told if your score is a new high (it will always be the new high after your first game) in a session. You are then asked if you want to play again. Press Y if you do, or N if you don't.

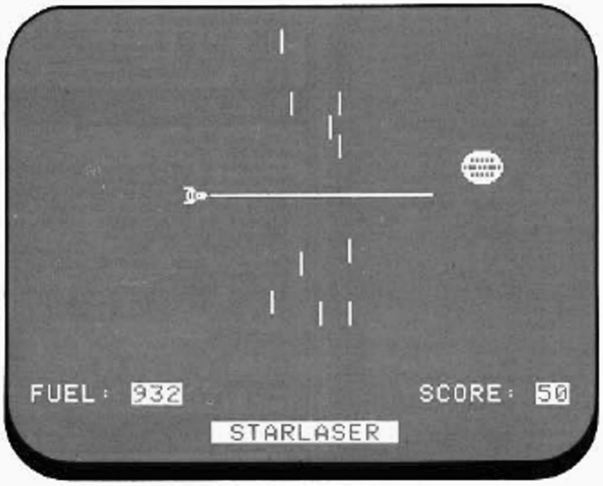

FIGURE 1: Starlaser Display

TABLE 1: Sound Table

Sound	Sequence	Pitch #1	Pitch #2	Delta #1	Delta #2	Duration
1	1	246	205	1	0	10
2	1	150	149	255	255	40
1	2	20	2	1	1	20
2	2	246	245	255	0	30
3	2	255	254	255	253	100
1	3	255	254	255	0	5
2	3	10	9	4	3	10
1	4	200	197	0	0	120
2	4	150	146	0	0	120

ENTERING THE PROGRAMS

To enter Starlaser, start by keying in **Listing 1** and saving it with the command:

```
SAVE STARLASER
```

Then enter the Monitor with CALL −151 and key in the hex code from **Listing 2**. Save the shape table with the command:

```
BSAVE STARLASER.SHPS,A$80B6,L$148
```

For the sound table, you can either use ASE from "Arcade Sound Editor" and the parameters from **Table 1**, or you can enter the Monitor with CALL −151 and key in the hex code from **Listing 3**. If you are using ASE, save the table with the name STARLASER .SNDS. If you key in the hex code, save the sound table with the command:

```
BSAVE STARLASER.SNDS,A$83BB,L$44
```

Finally, you need the program DUO from "Arcade Sound Editor." Follow the directions in that article for entering the code. Note that Starlaser uses the program in a different location, so if you need to correct errors, you should BLOAD the file at its original address and BSAVE it again according to the directions in "Arcade Sound Editor."

For help with entering *Nibble* listings, see "A Welcome to New *Nibble* Readers" at the beginning of this issue.

When you are done, your game disk should contain the files STARLASER, STARLASER.SHPS, STARLASER.SNDS and DUO. If Starlaser is unable to find one of these files, it will display the message "Wrong Disk or File(s) Missing."

HOW IT WORKS

Starlaser follows a standard game program organization and uses many of the programming techniques described in the book *Action Games for the Apple: How to Design Computer Games*, by S. Scott Zimmerman and Beverly B. Zimmerman, published by Scott, Foresman, and Company. The introduction (**lines 100-210** of **Listing 1**) verifies the existence of the binary files, prints the title and copyright notice for the program, BLOADs and BRUNs the binary files, sets up the arrays, and defines some of the initial variables. The key variables of the program are given in **Table 2**.

The setup section prepares the variables and the Hi-Res screen for the start of the game. **Line 250** POKEs in the address of the sound table, and **line 260** POKEs in the address of the shape table. (These POKEs are explained in "Arcade Sound Editor.") **Line 270** does a POKE 230,32 to specify that Hi-Res page 1 will be used; CALL 62450 clears the graphics screen without displaying it; and the other POKEs are to soft switches that specify the Hi-Res screen (−16297), mixed graphics and text (−16301), and Hi-Res page 1 (−16300). In **line 280**, the POKE −16304,0 effects the screen flip from text to graphics mode. The syntax &NORMAL in **line 280** is a DUO command that makes sure that the sound is turned on and is properly output to the built-in speaker.

Lines 290-360 initialize all of the variables, draw the shapes in their initial locations, print the initial fuel quantity and score, and make the warning sound.

The section of code labeled main loop (**lines 370-650**) is where all of the keyboard checking and animation takes place. During each turn of the main loop, each animation object moves by an incremental amount, and the fuel is decreased by a calculated amount.

Lines 400-470 check the keyboard, and if a key is pressed, make changes in the game accordingly. **Lines 480-500** make sure that the starship's rate of animation does not exceed 12 pixels per cycle of the main loop. They also calculate the fuel loss based upon the speed of the starship.

TABLE 2: Key Variables

Variable	Explanation
AB	Asteroid belt shape number (3)
AC	Asteroid counter (1-10); index of asteroids
AD	Calculated address for BRUNning DUO
AN	Asteroid number (10)
AS	Calculated address of sound table (STARLASER.SNDS)
AX	Calculated address of shape table (STARLASER.SHPS)
FC	Fuel count; amount of fuel used in each main loop cycle
FU	Fuel remaining (displayed on the screen)
HB( )	Variable in function to calculate high-order byte
HI	The highest address available for a binary file
HS	Current high score
KB	Kloryon base shape number (2)
LA	Laser beam shape number (4)
LB	Length of shape table (328 bytes)
LB( )	Variable in function to calculate low-order byte
MK	Movement value of Kloryon starbase
MS	Movement value of starship
OY	Old Y-coordinate for animation calculations
SC	Score of game
SF	Sound flag (1 if sound is on; 0 if sound is off)
SS	Starship shape number (1)
X	Temporary horizontal (X) coordinate of asteroid
XA( )	Horizontal coordinates of asteroids
XK	Horizontal coordinate of Kloryon base
XL	Horizontal coordinate of laser
XS	Horizontal coordinate of starship
Y	Temporary vertical coordinate of asteroid
YA( )	Vertical coordinates of asteroids
YK	Vertical coordinate of Kloryon base
YL	Vertical coordinate of laser
YS	Vertical coordinate of starship

Lines 510-540 animate the starship. The current vertical location (Y-coordinate) of the ship is saved in the variable OY (old Y), the new YS is calculated based on the movement variable (MS), and the screen boundaries are checked. **Line 540** erases the starship from its old location and redraws it in its new location.

Lines 550-560 increment the Kloryon count variable, KC. Whenever KC reaches 40 (40 times through the main loop), **line 560** calculates a (potential) new direction for the movement of the Kloryon starbase.

The animation of the Kloryon base takes place in **lines 570-600**, using basically the same algorithm as the animation of the starship. Animation of the asteroid barriers occurs in **lines 610-630**. Only one asteroid moves during each pass through the main loop. The variable AC (asteroid count) keeps track of which asteroid's turn it is to move.

Line 640 decreases the fuel during each cycle of the main loop, and **line 650** contains the GOTO to the start of the main loop.

Subroutines

Starlaser uses several short subroutines. **Lines 690-760** fire a laser after the Space bar is pressed. XL and YL are calculated from the location (XS,YS) of the starship. The laser shape is drawn on the screen, the laser sound (&1) is made, the collision between the laser and other objects (asteroids or Kloryon starbase) are determined, and the laser shape is erased.

Lines 770-800 include the subroutine that prints the current amount of fuel and the current score during the game. The subroutine also checks to see if the fuel has dropped below zero, which would end the game. It also makes sure that the score never goes below zero.

The subroutines in **lines 810-820** calculate a random location for the Kloryon starbase and a random direction of motion. The subroutines in **lines 830-840** calculate and print the new score and fuel level. Finally, the subroutine in **lines 850-860** produces the explosion sequence, sound effects, scoring, etc., when a Kloryon starbase is hit.

MODIFICATIONS AND ENHANCEMENTS

Some people may find the game too hard or too easy to play. The simplest way to change the level of difficulty is to change the number of asteroids. Just change the value of AN in **line 200**.

Another modification to make the game easier is to add the ability to stop the starship immediately, rather than having to decelerate to a stop. To do this, include the statement:

```
465 IF A$ = CHR$(13) THEN MS = 0
```

which will let you stop the starship by pressing Return.

You might want to make the game more fun and challenging by changing the rate at which the asteroids move down the screen, or making some asteroids move up while others move down. To change the rate of movement, modify **line 620** so that YA(AC) is incremented by 2, rather than 6 (to slow down the asteroids) or incremented by 12 (to speed up the asteroids).

Another modification that would make the game more personal is to design your own sound effects using "Arcade Sound Editor." You may also want to output the sound from your Apple II Plus or IIe to a stereo amplifier. Connect the Apple's cassette out port to the amplifier's audio in port, with your amplifier selected to tape. This necessitates changing the two occurrences of &NORMAL to &POP in **Listing 1** (in **lines 280 and 450**). If you have an Apple IIc, connect the earphone port directly to external speakers and do not modify **lines 280 and 450**.

While you're playing Starlaser, have a real blast!

LISTING 1: STARLASER

```
10  REM ************************
20  REM *                      *
30  REM *      STARLASER       *
40  REM * BY SCOTT ZIMMERMAN   *
50  REM * COPYRIGHT (C) 1987   *
60  REM * BY MICROSPARC, INC   *
70  REM * CONCORD, MA  01742   *
80  REM *                      *
90  REM ************************
100  REM * INTRODUCTION:
110  ONERR  GOTO 990
120  PRINT  CHR$ (4)"VERIFY STARLASER.SNDS": PRINT
      CHR$ (4)"VERIFY DUO": PRINT  CHR$ (4)"V
     ERIFY STARLASER.SHPS": POKE 216,0
130 HI = 33792: HIMEM: HI: TEXT : HOME
140  VTAB 5: HTAB 15: INVERSE : PRINT " STARL
     ASER ": VTAB 7: HTAB 10: NORMAL : PRINT
     "BY S. SCOTT ZIMMERMAN"
150  HTAB 12: PRINT "COPYRIGHT (C) 1987": HTAB
     13: PRINT "MICROSPARC, INC.": VTAB 20: HTAB
     14: PRINT "PLEASE ";: FLASH : PRINT "WAI
     T": NORMAL
160 L = 68:AS = HI - L - 1: PRINT  CHR$ (4);"
     BLOAD STARLASER.SNDS,A";AS
170 AD = AS - 445: PRINT  CHR$ (4);"BRUN DUO,
     A";AD
180 LB = 328:AX = AD - LB: PRINT  CHR$ (4);"B
     LOAD STARLASER.SHPS,A";AX
190 HI = 256 *  INT (AX / 256): HIMEM: HI: VTAB
     20: CALL  - 958: PRINT "PRESS <RETURN> T
     O BEGIN -> ";: GET A$
200 AN = 10: DIM XA(AN),YA(AN):HS = 0
210  DEF  FN HB(A) =  INT (A / 256): DEF  FN
     LB(A) = A -  FN HB(A) * 256:SF = 1
220  REM *----------------
230  REM * SETUP:
240  REM *----------------
250  POKE 206, FN LB(AS): POKE 207, FN HB(AS)
     : REM SOUND TABLE ADDRESS
260  POKE 232, FN LB(AX): POKE 233, FN HB(AX)
     : REM SHAPE TABLE ADDRESS
270  POKE 230,32: CALL 62450: POKE  - 16297,0
     : POKE  - 16301,0: POKE  - 16300,0
280  POKE  - 16304,0: IF SF THEN  &  NORMAL
290 SC = 0:FU = 1000:FC = 1:SS = 1:KB = 2:AB =
     3:LA = 4:MS = 0:KC = 0:AC = 0
300  HCOLOR= 3: SCALE= 1: ROT= 0
310  FOR I = 1 TO AN
320 X = 4 *  INT ( RND (1) * 12 + 28):Y =  INT
     ( RND (1) * 148)
330  XDRAW AB AT X,Y: IF  PEEK (234) < 10 THEN
      XDRAW AB AT X,Y: GOTO 320
340 XA(I) = X:YA(I) = Y: NEXT I
350  VTAB 21: HTAB 1: PRINT "FUEL:": GOSUB 77
     0: VTAB 21: HTAB 29: PRINT "SCORE:": GOSUB
     790: VTAB 23: HTAB 14: INVERSE : PRINT "
      STARLASER ": NORMAL
360 YS =  INT ( RND (1) * 150):XS = 80: XDRAW
     SS AT XS,YS: GOSUB 810: & 4: & 4
370  REM *----------------
380  REM * MAIN LOOP:
390  REM *----------------
400  IF  PEEK ( - 16384) < 128 THEN 510
410  GET A$
420  IF A$ =  CHR$ (8) THEN MS = MS - 1: GOTO
     480
430  IF A$ =  CHR$ (21) THEN MS = MS + 1: GOTO
     480
440  IF A$ = " " THEN  GOSUB 690: GOTO 510
450  IF A$ = "S" OR A$ =  CHR$ (115) THEN  &
      NORMAL :SF =  NOT SF: IF  NOT SF THEN  &
      STOP
```

LISTING 1: STARLASER *(continued)*

```
460  IF A$ =  CHR$ (27) THEN  WAIT  - 16384,1
     28: POKE  - 16368,0: GOTO 510
470  GOTO 510
480  IF MS <  - 12 THEN MS =  - 12
490  IF MS > 12 THEN MS = 12
500 FC =  INT ( ABS (MS) / 2) + 1
510 OY = YS:YS = YS + MS
520  IF YS < 0 THEN YS = 150
530  IF YS > 150 THEN YS = 0
540  XDRAW SS AT XS,OY: XDRAW SS AT XS,YS
550 KC = KC + 1: IF KC < 40 THEN 570
560 KC = 0: GOSUB 820
570 OY = YK:YK = YK + MK
580  IF YK < 0 THEN YK = 140
590  IF YK > 140 THEN YK = 0
600  XDRAW KB AT XK,OY: XDRAW KB AT XK,YK
610 AC = AC + 1: IF AC > AN THEN AC = 1
620  XDRAW AB AT XA(AC),YA(AC):YA(AC) = YA(AC
     ) + 6: IF YA(AC) > 148 THEN YA(AC) = 0
630  XDRAW AB AT XA(AC),YA(AC)
640 FU = FU - FC: GOSUB 770
650  GOTO 400
660  REM *----------------
670  REM * SUBROUTINES:
680  REM *----------------
690 XL = XS + 11:YL = YS + 3: XDRAW LA AT XL,
     YL: GOSUB 840: & 1
700 N = 0: IF  PEEK (234) = 118 THEN 760
710  FOR I = 1 TO AN: IF YA(I) > YL THEN 740
720  IF YA(I) + 9 < YL THEN 740
730 N = N + 1: GOSUB 830
740  NEXT I: IF N > 0 THEN 760
750  XDRAW LA AT XL,YL: GOSUB 850: RETURN
760  XDRAW LA AT XL,YL: GOSUB 790: RETURN
770  IF FU < 0 THEN FU = 0: GOSUB 780: POP : GOTO
     900
780  VTAB 21: HTAB 7: INVERSE : PRINT FU;: NORMAL
     : PRINT " "
790  IF SC < 0 THEN SC = 0
800  VTAB 21: HTAB 36: INVERSE : PRINT SC;: NORMAL
     : PRINT " ": RETURN
810  GOSUB 820:YK =  INT ( RND (1) * 140):XK =
     195: XDRAW KB AT XK,YK: RETURN
820 MK =  INT ( RND (1) * 2):MK = ( - 1) ^ MK
     : RETURN
830 SC = SC - 20: GOSUB 790: & 3: RETURN
840 FU = FU - 10: GOSUB 770: RETURN
850  XDRAW KB AT XK,YK: XDRAW 5 AT XK,YK: & 2
     ,1: XDRAW 5 AT XK,YK: XDRAW 6 AT XK,YK: &
     2,2: XDRAW 6 AT XK,YK: XDRAW 6 AT XK,YK +
     4: & 2,3: XDRAW 6 AT XK,YK + 4
860 SC = SC + 50: GOSUB 790: & 4: GOSUB 810: POKE
     - 16368,0: RETURN
870  REM *----------------
880  REM * END OF GAME:
890  REM *----------------
900  HOME : TEXT : VTAB 5: HTAB 12: INVERSE :
     PRINT " END OF MISSION ": VTAB 10: HTAB
     12: NORMAL : PRINT "YOUR SCORE: ";SC
910  VTAB 12: HTAB 12: PRINT "HIGH SCORE: ";H
     S
920  IF SC <  = HS THEN 940
930  VTAB 12: HTAB 12: CALL  - 958: FLASH : PRINT
     "NEW HIGH SCORE!": NORMAL : & 4,1: & 4,1
     : & 4,1:HS = SC
940  VTAB 20: CALL  - 958: & 3,1: PRINT "DO Y
     OU WANT TO PLAY AGAIN? (Y/N) ";: GET A$:
     PRINT A$
950  IF A$ = "Y" OR A$ =  CHR$ (121) THEN 250
960  IF A$ = "N" OR A$ =  CHR$ (110) THEN 980
970  GOTO 940
980  HOME : TEXT : END
990  HOME : VTAB 12: PRINT "WRONG DISK OR FIL
     E(S) MISSING"
1000  VTAB 22: PRINT "<ESC> TO QUIT, <RETURN>
     TO START OVER";: GET Z$: IF Z$ <  >  CHR$
     (27) GOTO 110
1010  END
```

END OF LISTING 1

```
KEY PERFECT 5.0
RUN ON
STARLASER

=====================================
CODE-5.0   LINE# - LINE#   CODE-4.0
--------   -------------   --------
7C62FD67     10 -   100      5176
DC45E772    110 -   200      EB99
1B034C95    210 -   300      96E0
AEFDECB0    310 -   400      88CE
F6382438    410 -   500      61F0
51D7367C    510 -   600      4E37
2A49C863    610 -   700      7587
E784244B    710 -   800      604D
CD2D8362    810 -   900      A355
CC572B4A    910 -  1000      A28F
B0516B70   1010 -  1010      0216
C85D7B12 = PROGRAM TOTAL =   0B21
```

LISTING 2: STARLASER.SHPS

```
80B6- 06 00
80B8- 0E 00 24 00 84 00 8A 00
80C0- C6 00 12 01 2D 75 3F 1E
80C8- 0D 0D 2D 75 3F 1F 1F 1F
80D0- 6E 0D 2D F5 DB 3F DE 2D
80D8- 35 00 49 09 2D 2D 2D 4E
80E0- 3F 3F 3F 3F 3F 1E 2D 2D
80E8- 2D 2D 2D 2D 0E 3F 1F 1F
80F0- 1F 1F 1F 3F 2E 6D 0D 0D
80F8- 0D 0D 2D 0E 3F 3F 3F 3F
8100- 3F 3F 3F 3F 6E 4D 69 0D
8108- 4D 69 FE DF FB 1F DF FB
8110- 2E 2D 2D 2D 2D 2D 2D 2D
8118- F5 3F 1F 1F 1F 1F 1F 3F
8120- 2E 6D 0D 0D 0D 0D 2D 1E
8128- 3F 3F 3F 3F 3F 3F 0E 2D
8130- 2D 2D 2D 2D DE 3F 3F 3F
8138- 06 00 36 36 36 36 36 00
8140- 2D 2D 2D 2D 2D 2D 2D 2D
8148- 2D 2D 2D 2D 2D 2D 2D 2D
8150- 2D 2D 2D 2D 2D 2D 2D 2D
8158- 2D 2D 2D 2D 2D 2D 2D 2D
8160- 2D 2D 2D 2D 2D 2D 2D 2D
8168- 2D 2D 2D 2D 2D 2D 2D 2D
8170- 2D 2D 2D 2D 2D 2D 2D 2D
8178- 2D 2D 2D 00 49 49 29 4D
8180- 4E 1F 3F 1F DF FB 0E 2D
8188- 4D 0D 6D 29 FE FB DB FB
8190- 1F 4E 0D 0D 4D 69 0D 0E
8198- FF 3B FF 3F 1F FF 4E 49
81A0- 0D 4E 1F DF DB DE 4D 0D
81A8- 6D 2D 6D 0E FF 1B 1F DF
81B0- F3 2B 6D 09 0D 0D 0D 0D
81B8- 1E 1F 3F 1F FF 1F 77 69
81C0- 6D 6D 1E FF 1F DF 33 00
81C8- 52 49 49 4D 0E FE DB DB
81D0- 2E 4D 69 DE 1B DE 1B 0E
81D8- 0D 6D 49 09 0E FF DB FB
81E0- 1B 4E 09 4D 4D 0D 1E 1F
81E8- DF DB DF F3 6B 09 4D 69
81F0- F1 DB DF 1B 4E 4E 49 49
81F8- DF FB F3 DB 33 00
```

END OF LISTING 2

LISTING 3: STARLASER.SNDS

```
83B8- 04 00 0A 00 17
83C0- 00 2A 00 37 00 02 F6 CD
83C8- 01 00 0A 00 96 95 FF FF
83D0- 28 00 03 14 02 01 01 14
83D8- 00 F6 F5 FF 00 1E 00 FF
83E0- FE FF FD 64 00 02 FF FE
83E8- FF 00 05 00 0A 09 04 03
83F0- 0A 00 02 C8 C5 00 00 78
83F8- 00 96 92 00 00 78 00
```

END OF LISTING 3

```
KEY PERFECT 5.0
RUN ON
STARLASER.SHPS

=====================================
CODE-5.0   ADDR# - ADDR#   CODE-4.0
--------   -------------   --------
22BA8129     80B6 - 8105      2725
A8AA95A3     8106 - 8155      2873
66567A3E     8156 - 81A5      276F
AD589481     81A6 - 81F5      2717
E4007440     81F6 - 81FD      03DD
FD9819D7 = PROGRAM TOTAL =    0148
```