Popular Electronics®

WORLD'S LARGEST-SELLING ELECTRONICS MAGAZINE JUNE 1980/95¢

First Look! RCA's New XL-100 Color TV

"Space Battle" Computer Program

Repel Small Animals with Harmless Pulses

Home Video Recorders Come of Age!

● How they work ● How to buy ● What's coming up

Carl Warren Talks About the "Source" and "MicroNet" Computer Time-Sharing Systems

JUNE 1980

video recorders

Toshiba and BASF. Neither system has a revolving head drum. Instead, they move the tape rapidly past a fixed head. To get reasonable recording times, both systems also use multiple tracks (300 for Toshiba, 72 for BASF).

Both systems also bear the same initials: LVR. But from that point on, all similarities (even what the initials stand for) cease. Toshiba's Longitudinal Video Recorder uses an endless loop of tape, running at 5.5 meters (18 feet) per second; the 135-meter loop runs through the recorder in only 24.6 seconds on each track. With 300 tracks, the total recording time is two hours.

In BASF's case, "LVR" stands for Linear Video Recorder. This system does not use an endless loop, but instead reverses at the end of each of its 72 tracks. With a tape length of 600 meters (1968 feet) and a speed of 4 meters (13.1 feet) per second, each track lasts 150 seconds, or two-and-one-half minutes. Track-switching time is a bit longer than Toshiba's (100 as opposed to 28 ms), but occurs only about one-sixth as often. While Toshiba's tape will measure the standard half inch (13 mm), BASF's will only be 8 mm wide.

Both systems have advantages over helical-scan systems: mechanisms can be simpler (Toshiba claims a one-third saving in parts), and cost lower (Toshiba hints at recorders for about $500, and playback-only machines for about $300). Access to any part of the tape is quicker, too: Toshiba cites a maximum access time of 8.4 seconds to get from the first to the last track of its LVR, with 33 seconds the worst possible access time to any point on the tape. Both also open the possibility of cheaper commercially recorded cassettes. Tape costs will be low and duplicators will be able to make copies in far less time than required for helical-scan tapes.

BASF's LVR, already sold in Europe, will be built in a California plant now in operation, and should be on the market soon. Toshiba's will probably be available late this year. Toshiba also hopes to have the ultimate video portable—a camera with recorder built in—in about 2 years. As BASF's cassette is even smaller, they're likely to come out with a similar product.

But the smallest tape cassette is probably Funai's. Its CVC (Compact Video Cassette) system records for 30 minutes, and uses quarter-inch tape, so its cassette is only about one-fourth the BASF cassette's size, and its player is about the size and weight of a large portable audio cassette recorder. Technicolor may import it into the U.S. Other than that it's a helical-scan system, few details have yet been released. ◇

PLAY "SPACE BATTLE" ON YOUR VIDEO MONITOR

Here's an exciting visual game program for computers using an RCA COSMAC 1802 CPU

BY DONALD R. SCHROYER

THIS program for 1802-based systems equipped with an 1861 video display, puts you and your opponent in command of a pair of space vehicles. The ships move in accordance with commands entered via the hex keypad, and fire their weapons via a pair of pushbutton switches tied to input lines EF3 and EF4. When a player has scored eight hits against his opponent, the losing ship is displayed as destroyed. At the end of the game, the score is displayed. About 2K of memory is required.

The Program. This consists of an initialization routine, six subroutines, the main program, six data pointers and six data locations. These are shown in Tables I through IX.

The Input and Movement subroutine shown in Table I inputs data via the hex keypad. Depressing keys 0, 4, 8, or C results in movement of the left ship. Keys 0 and 4 move the ship downward with 0 causing motion at twice the speed of key 4. Keys 8 and C move the left ship up with the rate of movement doubled when key C is depressed. Keys 1, 5, 9, or D freeze the left ship where it is.

Keys 3, 7, B, or F operate the right

ship in a like manner. Keys 2, 6, A, or E freeze the right ship. If your system has a hex display, the last two movements will be displayed.

In operation, movement of the two ships is accomplished by changing R(7) the left ship position pointer, and R(8) the right ship position pointer. The ships are moved up by subtracting 08 from the pertinent position pointer. Downward motion is performed by adding 08 to the pointers. To double the speed, 10 is added or subtracted. The subroutine tests the last position of each ship to assure that neither moves out of the screen's display area.

The Load Counter R(5) subroutine of Table II replicates the data pointed to by R(3) at the display location pointed to by R(7). The data pointed to by R(3) forms the silhouette of the left ship. The silhouette of the right ship, pointed to by R(4) is loaded into the display area pointed to by R(8).

The Left Fire Counter R(A) subroutine of Table III tests the switch wired to $\overline{EF4}$. If this line is low, the data flag is set and R(9) is called, otherwise the main program continues. The Right Fire Counter R(B) subroutine of Table IV tests $\overline{EF3}$ in the same manner.

The Weapons Counter R(9) subroutine of Table V is brought into play only when called by subroutines R(A) or R(B). When entered, the Weapons subroutine tests the data flag and fires either the left or right ship's weapon as appropriate. The weapons of both ships cannot be fired simultaneously, and this is compensated for in the main program. The Weapons subroutine also tests each firing to see if a hit has been scored. If so, the score stored in R(E).1 is incremented by adding 10 for a left hit and 01 for a right hit. Register R(E).1 stores the left score in the high four bits and the right score in the low four bits. The Weapons subroutine tests to see if either score is equal to eight. If so, the game ends with the score displayed, otherwise the program continues.

The Interrupt Counter R(1) shown in Table VI displays a two-page segment of memory in a 64 x 64 format. This display area is from 0400 to 05FF.

Table VII, Main Counter R(F), is a sequence of calls to the various subroutines. Since the Weapons subroutine cannot simultaneously fire both ship's weapons, the left and right fire subroutines alternate. Once the program is running, the alternation gives each player roughly an even chance of firing first if both fire switches are operated simultaneously. The main program inserts a delay routine between each call to R(C) and R(5) to slow the ship's movements for easier control.

TABLE I—INPUT AND MOVEMENT COUNTER R(C)

```
0200 DF
0201 F8 06 B6 A6 E6
0206 6C FA F0 AE
020A FA 20 32 0F 16 16
0210 8E F6 F6 F6 F6 56
0216 F8 06 A6
0219 72 FA 0F AD
021D FA 02 32 22 16
0222 8D 56 F8 06 A6 8D 56
0229 8E F4 56 64
022D F0 32 55
0230 F0 FB 03 32 69
0235 F0 FB 0C 32 7D
023A F0 FB 0F 32 91
023F F0 FB 04 32 A5
0244 F0 FB 07 32 B9
0249 F0 FB 08 32 CD
024E F0 FB 0B 32 E1
0253 30 F3
0255 97 FB 05 3A 5F
025A 87 FC 58 33 F3
025F 87 FC 10 A7
0263 97 7C 00 B7
0267 30 F3
0269 98 FB 05 3A 73
026E 88 FC 51 33 F3
0273 88 FC 10 A8
0277 98 7C 00 B8 30 F3
027D 97 FB 05 32 87
0282 97 FB 03 32 F3
0287 87 FF 10 A7
028B 97 7F 00 B7 30 F3
0291 98 FB 05 32 9B
0296 98 FB 03 32 F3
029B 88 FF 10 A8
029F 98 7F 00 B8 30 F3
02A5 97 FB 05 3A AF
02AA 87 FC 50 33 F3
02AF 87 FC 08 A7
02B3 97 7C 00 B7 30 F3
02B9 98 FB 05 3A C3
02BE 88 FC 49 33 F3
02C3 88 FC 08 A8
02C7 98 7C 00 B8 30 F3
02CD 97 FB 05 32 D7
02D2 97 FB 03 32 F3
02D7 87 FF 08 A7
02DB 97 7F 00 B7 30 F3
02EI 98 FB 05 32 EB
02E6 98 FB 03 32 F3
02EB 88 FF 08 A8
02EF 98 7F 00 B8
02F3 86 FB 07 3A FB
02F8 16 30 2D
02FB E2 30 00
```

TABLE II—LOAD COUNTER R(5)

```
014F DF
0150 83 A6 93 B6
0154 F8 00 AE
0157 97 73 87 52
015B 46 57 1E
015E 8E FB 0A
0161 32 6D
0163 87 FC 08 A7
0167 97 7C 00 B7
016B 30 5B
016D 72 A7 F0 B7
0171 84 A6 94 B6
0175 F8 00 AE
0178 98 73 88 52
017C 46 58 1E
017F 8E FB 0A
0182 32 8E
0184 88 FC 08 A8
0188 98 7C 00 B8
018C 30 7C
018E 72 A8 F0 B8
0192 30 4F
```

TABLE III—LEFT FIRE COUNTER R(A)

```
0194 DF
0195 3F 94 F8 01 F6
019A D9 30 F1 00 00
```

TABLE IV—RIGHT FIRE COUNTER R(B)

```
019F DF
01A0 3E 9F F8 00 F6
01A5 D9 30 F6 00 00
01F0 DF 3F 94 30 F0
01F5 DF 3E 9F 30 F5
```

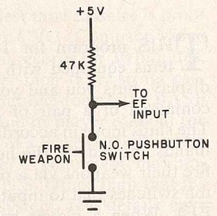

Fig. 1. To implement the space battle program, the circuit shown here must be duplicated, with one attached to $\overline{EF3}$ and the other connected to $\overline{EF4}$.

TABLE V—WEAPONS COUNTER R(9)

```
0300 DF
0301 7A 7B
0303 33 07
0305 30 79
0307 87 FC 19 A6 97 7C 00 B6
030F 96 73 86 52 F8 00 AE E6
0317 F8 FF 56 16 1E
031C 8E FB 06 3A 17
0321 F8 FF F2 E2 3A 2C
0327 F8 FF 56 30 5C 22
032D 94 73 84 73
0331 F8 10 A4 F8 06 B4
0337 F8 00 AE 98 73 88 52
033E 44 58 1E 8E FB 0A
0344 32 50
0346 88 FC 08 A8 98 7C 00 B8
034E 30 3E 9E FC 10 BE
0354 72 A8 72 B8 72 A4 72 B4
035C 72 A6 F0 B6 F8 00 AD BD
0364 1D 9D FB 01 3A 64
036A F8 00 AE F8 00 56 16 1E
0372 8E FB 07 3A 6D 30 E8
0379 88 FC 27 A6 98 7C 00 B6
0381 96 73 86 52 F8 00 AE E6
0389 F8 FF 73 1E 8E FB 06
0390 3A 89
0392 F8 FF F2 E2 3A 9D
0398 F8 FF 56 30 CD 22
039E 93 73 83 73
03A2 F8 10 A3 F8 06 B3
03A8 F8 00 AE 97 73 87 52
03AF 43 57 1E 8E FB 0A 32 C1
03B7 87 FC 08 A7 97 7C 00 B7
03BF 30 AF 9E FC 01 BE
03C5 72 A7 72 B7
03C9 72 A3 72 B3
03CD 72 A6 F0 B6
03D1 F8 00 AD BD 1D
03D6 9D FB 01 3A D5
03DB F8 00 AE
03DE F8 00 56 26 1E
03E3 8E FB 07 3A DE
03E8 9E FA 88 32 00
03ED C0 01 C0
```

TABLE VI—INTERRUPT COUNTER R(1)

```
01CE 72 70
01D0 C4 22 78 22 52
01D5 F8 04 B0 F8 00 A0
01DB C4 C4 E2
01DE 80 E2 20 A0 E2
01E3 3C DE
01E5 80 E2 20 A0 E2
01EA 34 E5 30 CE
```

TABLE VII—MAIN COUNTER R(F)

```
0650 3F 50 37 52 D5 DC
0656 F8 00 AD BD 1D
065B 9D FB 02 3A 5A
0660 D5 DA DB DC
0664 F8 00 AD BD 1D
0669 9D FB 02 3A 68
066E D5 DB DA
0671 30 55
```

TABLE VIII—INITIALIZATION

```
0100 71 00
0102 F8 06 B2 BD BF
0107 F8 05 B7
010A F8 04 B8
010D F8 03 B9
0110 F8 02 BC
0113 F8 01 B1 B3 B4 B5 B6
011A BA BB A9 AC
011E F8 D0 A1 F8 3F A2
0124 F8 AA A3 F8 B4 A4
012A F8 50 A5 AF
012E F8 CC A6
0131 F8 C0 A7
0134 F8 07 A8 AD
0138 F8 95 AA F8 A0 AB
013E F8 00 BE
0141 F8 01 5D 1D
0145 F8 02 5D
0148 E6 69 E0
014B 70 2F
```

TABLE IX—DATA POINTERS

Left Ship
01AA 00 00 F2 F7 24
01AF 28 78 38 00 00
Alternate (see text)
01AA 00 00 18 FF 3C
01AF 18 3C FF 00 00
Right Ship
01B4 00 00 06 FF CE
01B9 86 0F 0F 00 00
Alternate (see text)
01B4 00 00 03 3F 7F
01B9 D4 3E 3E 00 00
Blast
0610 28 04 50 99 4E
0615 39 88 34 45 A2
End of Program
01C0 F8 06 A6 B6 E6
01C5 9E 56 64 26 30 C9
Display Area
0400 - 05FF; 00 loaded in display ; memory locations

Note: *A cassette tape of the program for Space Battle is available from Donald R. Schroyer, 209 Brinker St., Latrobe, PA 15650, for $10.00.*

In addition to the mentioned pointers, register R(6) serves as a temporary pointer to save and increment other pointers. The stack pointer is R(2) with the stack starting at 063F.

Memory location 0607 contains the current move instruction for the left ship, while 0608 contains the same data for the right ship. At Initialization (Table VIII), the contents of these two locations are 01 and 02, respectively. As stated earlier, R(E).1 holds the current score and is loaded with 00 on initialization. R(E).0 is an index counter used in loading the ship silhouettes and firing the weapons. R(D) is used to load 0607 and 0608 and for delays encountered in the main program.

Table IX contains the remainder of the program. Both the left and right ship are illustrated. Location 0610 to 0619 contains the blast that occurs after every hit and the debris of the destroyed ship. If the ship is moving fast enough at the time of a hit, debris is shown scattered in its wake.

Hardware. The only hardware connection required is a pair of circuits similar to that shown in Fig. 1, with one connected to $\overline{EF3}$ and the other to $\overline{EF4}$. These are the weapon firing switches.

Playing the Game. Load the program from Tables I through IX, making sure that the correct code is placed at the correct addresses. Since the 1802 starts running at address 0000, key in C0 01 00 starting at address 0000. Save the program on cassette if you have this provision.

To begin the game, reset and run the computer, then depress and release the INPUT switch on an ELF II, or bring the $\overline{EF4}$ line low then high. The two ships will appear in opposite diagonal corners.

The left ship cannot move until the right player has made a move. Once the right player has moved, the left ship will move according to the keypad data last entered. To fire the weapon, depress the appropriate FIRE WEAPON pushbutton switch. When the program is restarted after eight hits, each ship can be moved across the screen to remove "ghosts".

Program Changes. If you desire another look for the spaceships, try the alternate versions shown in Table IX. You can even design your own ships using graph paper. Make the first and last two bytes 00 or the ships will leave a trail as they move.

The winning score can be 1,2,4 or 8 as determined by address 03EA. To change ship speeds, locations 065D and 066A can be loaded with higher numbers to slow the ships down. If you desire more speed, use 01 or NOP's. ◇