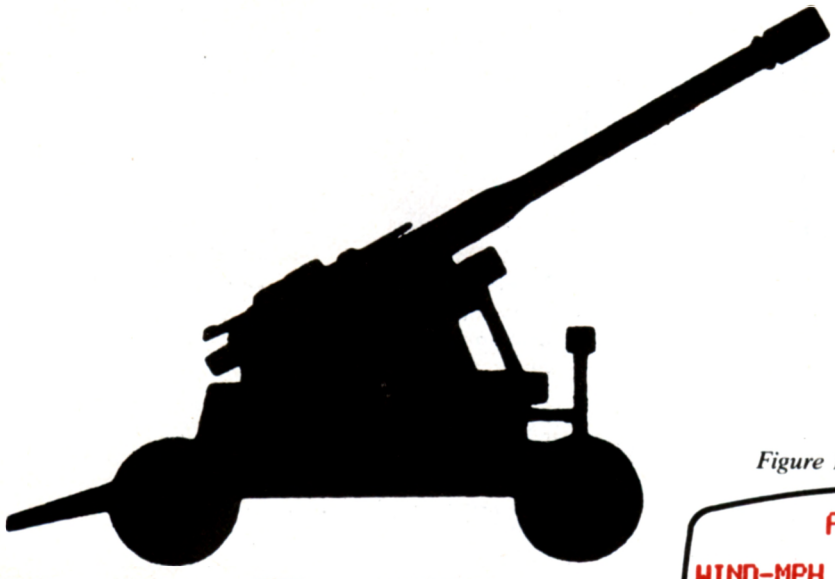


Bally Artillery

John W. Rhodes

In December of 1978 I was ready to buy my first computer system, but my requirements were not easy to meet. I wanted something that could handle arcade-quality games, had high-resolution graphics capability, color display, and Basic in PROM. I was not satisfied with anything my local dealers had to show (no one I visited had a Compucolor, the Apple dealers were showing low-resolution only, and the Atari was only a rumor), but on the basis of the (somewhat premature) advertising for the keyboard/expansion unit, I decided to buy a Bally Professional Arcade. I could use Tiny Basic for a while, and turn it into a "real" machine in just a few short months.

It was just a few short months later that the local dealers began to show Compucolors and high-resolution Apples, and it seemed that the Bally expansion unit was more of a rumor than the Atari 800. I would visit the showrooms, see those beautiful full-size keyboards, watch people work in "real" Basic and be as green as the color monitors.

I particularly liked the artillery game that Compucolor called "Shoot." This game generates a random terrain display and wind factor and positions two artillery emplacements on the screen so that two opponents can take turns trying to obliterate each other. Eventually I resolved that I either had to buy a Compucolor or program this game on my Bally. I chose the latter.

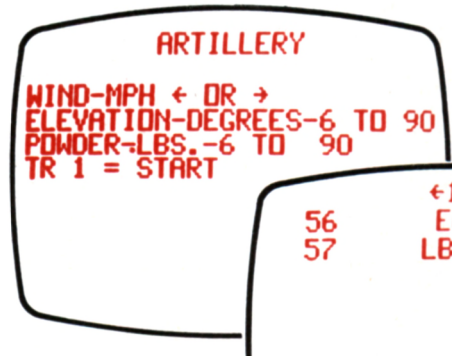
This turned out to be quite a challenge with less than 2K of memory and integer-only Tiny Basic. But the Bally Basic is quite sound for game programming and easy to work with. The greatest difficulty was finding an integer sine routine, but after searching the magazines I found a routine to adapt to my purpose. I started out using a full ballistic equation, but soon found by experimentation that I could use an approximation. This eliminated an integer square-root routine and added speed in the bargain.

I spent approximately two months writing, debugging, and fine-tuning the program, but it was worth the effort.

A few months later I did buy the Compucolor and have been using it ever since. I'm well satisified with it and use it for a variety of tasks. But my wife and I still enjoy the Bally for its games, especially the artillery game.

John W. Rhodes, Box 592, Belmont, CA 94002.

Figure 1. Player instructions.



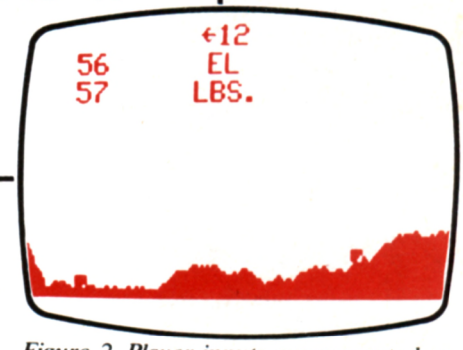


Figure 2. Player inputs are prompted on screen. Player sees values on screen as knob is turned. Pulling trigger enters value to program.

Explanation of the Program

Lines 10-28	print the introduction and wait for the player to press the trigger.
Lines 40-88	generate random terrain features and gun emplacements.
Lines 90-94	set and display the wind velocity.
Lines 100-140	input the elevation via knob, print the elevation and record when the trigger is pressed. They also calculate relative sine and cosine.
Lines 150-182	input powder via knob, print and record when the trigger is pressed. They also clear prompters and print elevation and powder values on the player's side of the screen.
Lines 190-306	compute shell movement and decide if a shell has 1) hit the target, 2) hit the ground, or 3) moved off the screen (left or right). If none of the above, a dot is printed and its position recorded.
Lines 400-408	clear the dots to prepare for the next player's turn.
Lines 500-514	make the explosion and wait for the trigger input to start the next game.
Lines 600-612	make a shell crater in ground.

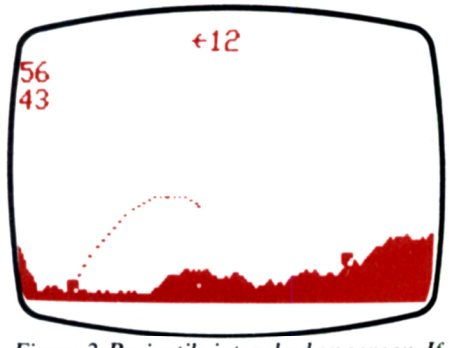
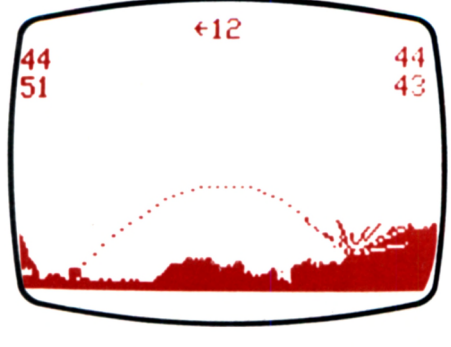


Figure 3. Projectile is tracked on screen. If the projectile impacts the ground (or goes off screen) the track is erased before the next player's turn.

Bally Artillery Listing by John W. Rhodes.
Note: Parenthetical comments are not part of the program.

```
10  CLEAR                          (Introduction)
12  PRINT"          ARTILLERY"     (9 spaces)
14  PRINT
16  PRINT" WIND-MPH (OR)"          (1 space)
20  PRINT" ELEVATION-DEGREES-6 TO 90"
22  PRINT" POWDER-LBS.-6 TO 90"
24  PRINT" TR 1 = START"
26  IF TR(1)=1 GOTO 40
28  GOTO 26
40  M=1                            (Start with Player 1)
50  CLEAR:FC=165:BC=9              (Generate Terrain)
52  B=RND(40):B=-B
54  I=RND(40)-70:V=RND(40)+30
60  FOR A=1 TO 4
62  C=fix40-120:D=RND(3)
64  FOR E=1 TO 40
66  H=RND(3):IF D=1 B=B+H
68  IF D=2 B=B+H-2
70  IF D=3 B=B-H
72  IF B>-5 B=-5:D=2
74  IF B<-42 B=-42:D=2
76  C=C+1:LINE C,-44,4:LINE C,B,1
78  IF C=I J=B
80  IF C=V K=B
82  NEXT E
84  NEXT A
86  BOX I,J+2,4,4,3                (Fix Gun Emplacement)
88  BOX V,K+2,4,4,3
90  L=RND(61)-31                   (Generate Wind)
92  CX=-8:CY=40:IF L>0 PRINT" ",@0,L,@0,"→"
94  IF L<0 PRINT"←",@0,-L
100 Z=0:B=512:R=1:E=1          (Initialize Sine Routine)
102 IF M=-1 R=2                    (Start Input Routine)
104 CX=-5:CY=31:PRINT"EL";NT=0
106 C=KN(R)÷6+24                   (C is Elevation)
108 IF M=1 CX=-53
110 IF M=-1 CX=45
112 CY=31:PRINT @0,Cx2
114 IF TR(R)=1 GOTO 120
116 GOTO 106
120 NT=3
```

Figure 4. The aftermath of a startling explosion (complete with bells and whistles).



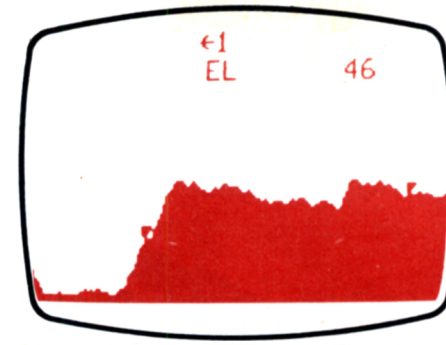


Figure 5. Start of next game. Terrain and wind factors differ dramatically from game to game.

```
130 FOR D=1 TO C                   (Sine Routine)
132 Z=Z+(B÷28):B=B-(Z÷28)
134 NEXT D
136 IF Z>512 Z=512
138 IF B<0 B=0
140 Z=Z÷10:B=B÷10
150 CX=-10:CY=22:PRINT"LBS.";NT=0
152 F=KN(R)÷3+48                   (F is Powder)
154 IF M=1 CX=-53
156 IF M=-1 CX=45
158 CY=22:PRINT @0,F
160 IF TR(R)=1 GOTO 170
162 GOTO 152
170 IF M=1 CX=-72
172 IF M=-1 CX=65
174 CY=31:PRINT @0,Cx2
176 IF M=1 CX=-72
178 IF M=-1 CX=65
180 CY=22:PRINT @0,F
182 BOX 0,26,120,20,2:NT=3:F=Fx4   (Clear Prompters)
190 G=ZxF÷51:H=BxF÷51              (Velocity Vectors)
192 W=0:X=0:H=HxM:E=1              (Initialize Movement)
200 U=(HxL)÷800:U=UxM              (Movement Routine)
202 W=W+(H÷4)+(U÷4)
204 X=X+(G÷4)-2
206 H=H+U:G=G-8:P=W÷10
208 IF M=1 P=P+I
210 IF M=-1 P=P+V
212 Q=X÷10:IF M=1 Q=Q+J+2   (What to do with position)
213 IF M=-1 Q=Q+K+2
214 IF P<-80 GOTO 400
216 IF P>80 GOTO 400
218 IF Q<-43 GOTO 400
220 IF M=1 IF P>(V-4) IF P<(V+4) GOTO 226
222 IF M=-1 IF P>(I-4) IF P<(I+4) GOTO 226
224 GOTO 230
226 IF M=1 IF Q>(K-4) IF Q<(K+4) GOTO 500
228 IF M=-1 IF Q>(J-4) IF Q<(J+4) GOTO 500
230 IF Q>-4 GOTO 300
232 IF PX(P,Q)=1 GOTO 600
300 MU=2:IF Q>44 GOTO 200          (Paint Dot Routine)
302 BOX P,Q,1,1,3
304 @(E)=P:@(E+1)=Q:E=E+2          (Remember Dot Position)
306 GOTO 200
400 M=-M                           (No Hit. Clear Dots)
402 FOR S=1 TO E-1 STEP 2
404 BOX @(S),@(S+1),1,1,3
406 NEXT S
408 GOTO 100
500 FOR N=-5 TO 5                  (Hit. Make Explosion)
501 FC=RND(32)x8+4:BC=FC-4
502 IF M=-1 K=J:V=I
504 LINE V,K,4
506 LINE NxRND(5)+V,K+RND(10),3:MU=1
508 NEXT N
510 M=-M                           (Switch Players)
512 IF TR(1)=1 GOTO 50
514 GOTO 512
600 IF M=1 IF P<I+3 GOTO 200   (Shell Impacted Ground)
602 IF M=-1 IF P>V-3 GOTO 200
604 FOR T=1 TO 5
606 MU=4
608 BOX P+T-3,Q+1,1,RND(4),2
610 NEXT T
612 GOTO 400
```