

General Information

Author: Dan Bullok

Language: ACTION!

Compiler/Interpreter: ACTION!

Published: Analog #28 (03/ 85)

Demon Birds#

You are the last Wizard of Akturnis, the strange and mystical world where magic can be worked by anyone with the will to do so. But, in the past few years, people have lost their faith in Wizards and magic. Now the evil **Demon Birds** have begun to plague them, and you are their *only* hope.

To save the People of Akturnis, you must enter the dreaded Valley of Deatz and destroy all of the **Demon Birds** found there.

Your Wizard starts the game with four lives and fifty units of energy. For every bird you destroy, you will gain two units of energy. However, every time you cast a fireball, you *lose* one unit of energy.

You move your Wizard left and right at the bottom of the screen, using the joystick. You may cast a fireball by pressing the red button while moving in the direction in which you wish it to travel.

Ridding your people of the **Demon Birds** will not be easy. If you are struck by one of the evil birds, or are hit by a meteor from the sky, you will lose one life. You'll also lose a life if your energy reaches zero. Furthermore, the ground in the valley is very unstable, because it sits on top of a pool of lava. If you stand in one place for too long, the ground will open up, and your Wizard will be lost.

Disk instructions.#

1. Type in Listing 1 and SAVE it to disk under the filename "D:BIRDS". You must have 48K and the Action! cartridge.
2. Reboot your computer and enter the monitor. Type C "BIRDS?".
3. When the disk drive stops, type W "AUTORUN.SYS" to save the object code to disk.
4. Whenever you want to play **Demon Birds**, insert the Action! cartridge into the left slot. Insert the disk with the AUTORUN.SYS file into drive one and turn on the computer. The program will load and run automatically.

Cassette instructions.#

1. Type in Listing 1 and SAVE it to cassette. You must have at least 48K and the Action! cartridge.
2. Reboot your computer and enter the monitor. Type C "C:".
3. When the cassette stops, type W "C:" to save the object code to cassette.
4. Whenever you want to play **Demon Birds**, insert the Action! cartridge into the left slot. Insert the cassette with the object code into the cassette recorder. Turn on the computer and enter the monitor. Type R "C:". The program will load and run automatically.

That's all there is to it. You're ready to do battle with the **Demon Birds**.

```
;*****  
;*          *  
;* Demon Birds *  
;*      by      *  
;*  Dan Bullok  *  
;*          *  
;*
```

```
;*
;*****
```

```
;Data For Player 0
```

```
BYTE ARRAY p0=[12 12 12 12 4 12 14 30
 29 45 13 0 0 0 0 0 0 0 102 12 12
 12 12 4 12 14 14 13 30 12 0 0 0 0 0
 0 0 2 50 12 12 12 12 4 12 14 14 14
 30 12 0 0 0 0 4 4 0 24 12 12 12
 12 4 12 12 12 14 14 28 0 0 0 0 0 0
 0 48 6 48 48 48 32 48 112 120
 184 180 176 0 0 0 0 0 0 0 102 48
 48 48 48 32 48 112 112 176 120 48 0
 0 0 0 0 0 64 76 48 48 48 48 32 48
 112 112 112 120 48 0 0 0 0 32 32
 0 24 48 48 48 32 48 48 48 112
 112 56 0 0 0 0 0 0 12 96]
```

```
;Data For Player 1
```

```
BYTE ARRAY p1=[0 0 0 0 4 12 14 30 29
 13 12 12 28 28 20 50 34 34 34 102 0
 0 0 0 4 12 14 14 13 14 8 12 12 12
 28 24 20 20 18 50 0 0 0 0 4 12 14
 14 14 14 8 12 12 28 28 8 12 12 8 24
 0 0 0 0 4 12 12 12 14 14 12 12 12
 12 12 20 20 18 50 6 0 0 0 32 48
 112 120 184 176 48 48 56 56 40 76
 68 68 68 102 0 0 0 0 32 48 112 112
 176 112 16 48 48 48 56 24 40 40 72
 76 0 0 0 0 32 48 112 112 112 112 16
 48 48 56 56 16 48 48 16 24 0 0 0 0
 32 48 48 48 112 112 48 48 48 48 48
 40 40 72 76 96]
```

```
;Meteor Data
```

```
BYTE ARRAY
 ball=[60 126 255 255 255 255
 126 60],
 ball2(8),coordstore(30)
```

```
;Character Set
```

```
BYTE ARRAY chset=
 [0 0 0 0 0 0 0 0
 0 32 32 160 168 168 170 170
 170 170 170 170 170 170 170 170
 0 128 128 128 162 170 170 170
 128 128 128 136 136 168 170 170
 0 0 0 0 0 0 170 170
 0 0 2 34 42 170 170 170
 0 2 2 2 34 42 42 170
 0 0 0 2 2 34 42 170
 0 5 85 1 1 1 0 0
 20 92 85 64 64 64 64 0
 0 1 1 1 5 85 0 0
 64 64 64 84 92 85 0 0
 20 53 85 1 1 1 1 0
 0 80 85 64 64 64 0 0
 1 1 1 21 53 85 0 0
 0 64 64 64 80 85 0 0]
```

```
252 254 102 102 102 254 252 0
0 0 60 102 124 96 56 14
0 0 254 255 219 219 219 3
0 0 63 102 102 102 60 0
0 0 220 102 102 102 246 7
252 254 102 126 102 254 252 0
24 0 56 24 24 24 62 0
0 0 223 96 96 96 240 0
14 12 252 204 204 204 119 0
0 0 62 96 60 6 252 0
0 195 60 60 60 195 0 0],
```

chset2=

```
[15 31 63 120 120 120 124 127
248 252 252 28 28 28 28 220
127 124 124 126 126 126 126 60
220 28 28 28 28 20 20 8
126 126 127 127 121 121 124 124
8 28 28 28 156 156 220 220
126 126 127 127 127 127 127 62
252 124 124 60 60 20 20 8
96 112 112 112 112 112 112 120
0 0 0 0 0 0 0 0
120 124 124 126 126 127 127 63
0 0 0 0 0 248 236 248
63 127 127 120 112 112 112 120
248 252 252 28 28 28 28 28
120 124 124 126 126 127 127 63
28 28 28 28 28 244 244 248
63 127 127 120 112 112 112 121
248 252 252 0 0 0 0 248
120 124 124 126 126 127 127 63
252 124 28 28 28 244 244 248]
```

;Notes for song

BYTE ARRAY notes=

```
[243 243 162 182 162 182 193 243],
```

notes1=

```
[162 96 108 121 108 121 128 162],
```

dur=[10 10 30 6 6 6 10 20],

increase=[2 0]

;y-positions of birds

BYTE ARRAY strafey=

```
[10 11 12 13 14 15 16 17 18 19 19
19 19 18 17 16 15 14 13 12 11 10
11 12 11 10 11 11 10 10 10 10 10
10 10 10 10 10 10 10 07 08 10 12
14 16 17 17 18 18 18 18 18 17 16
15 15 14 14 13 13 13 13 14 14 14
15 15 14 13 12 12 11 10 10 10 09
08 08 08 14 14 14 15 15 15 16 16
17 18 18 18 18 18 18 18 18 18 19
19 20 20 20 20 20 20 19 19 19 18
18 17 17 16 15 15 14 14 14 14 12
12 13 14 15 16 18 19 19 20 20 19
19 18 16 15 14 13 12 12 13 13 14
15 16 17 18 18 18 18 18 18 18 16
15 14 13 12 12 12 14 14 14 14 15]
```

```
16 16 17 18 18 19 19 19 19 19 18
18 17 17 17 17 17 17 18 18 19 19
19 19 19 18 17 17 16 16 15 15 14
14 14 14 14 15 16 16 16 17 17 18
18 18 19 19 19 20 20 19 19 19 19
18 18 17 17 16 16 16 15 15 15 15
15 15 14 14 14 14 14 14 14]
```

```
BYTE ARRAY flapinc=[1 0], bexist(10)
```

```
BYTE bcount,char1,char2,dieflag,bx,
by,fallx,fally,fallflag,bflap
```

```
;Miscellaneous variables
```

```
BYTE a,b,c,d,e,x=[100],y=[154],
ctr=[0],dir,fx,fy,fireflag,df,
mx=[10],my=[10],chad,men=[4],
memory,gflag=[1]
```

```
;Hardware registers
```

```
BYTE vcount=54283,colpf0=53270,
colpf1=53271,colpf2=53272,
colpf3=53273,wsync=54282,
chbase=54281,random=53770,
consol=53279,rtclock=20,ch=764
```

```
CARD pmbase,ac,bc,cc,vdslst=512,
dlilvec,score=[0],energy=[50]
```

```
PROC Dli2()
```

```
;Changes color of text window to red
```

```
[72 169 68 141 10
212 141 24 208
169 0 141 23 208]
```

```
vdslst=dlilvec
```

```
[104 64]
```

```
RETURN
```

```
PROC Dli1()
```

```
;Changes color of ground to Brown
```

```
[72 169 20 141 10
212 141 23 208]
```

```
vdslst=Dli2
```

```
[104 64]
```

```
RETURN
```

```
INT FUNC DeltaX()
```

```
;Returns Delta-X value of stick(0)
```

```
BYTE aa
```

```
INT xx
```

```
aa=Stick(0)
```

```
IF aa>12 THEN xx=0
```

```
ELSEIF aa<8 THEN xx=1 dir=80
ELSE xx=-1 dir=0
FI
RETURN(xx)
```

```
PROC Center(CARD cnum
            BYTE basx,basy)
```

```
;right-justifies number
IF cnum<10 THEN
    Position(basx,basy)
    PrintD(6,"0")
ELSEIF cnum<100 THEN
    Position(basx-1,basy)
    PrintD(6,"0")
ELSEIF cnum<1000 THEN
    Position(basx-2,basy)
    PrintD(6,"0")
ELSE
    Position(basx-3,basy)
    PrintD(6," ")
FI
PrintCD(6,cnum)
RETURN
```

```
PROC Delay(CARD cnt)
;Delay Loop
```

```
CARD cnnt
```

```
FOR cnnt=1 TO cnt DO OD
RETURN
```

```
PROC PMove(CARD pm,add
           BYTE plr,px,py,pix)
```

```
;Moves Player
;Variables passed:
;pm: address of pmbase
;add: address of source image
;plr: # of player to move 0-3
;px: x-position of player
;py: y-position of player
;pix: number of bytes to move
```

```
px==+48
py==+32 ;add screen margin offsets
ac=pm+1024+plr*256 ;add work space
Zero(ac+py-5,pix+10) ;clear area out
MoveBlock(ac+py,add,pix)
Poke(53248+plr,px)
RETURN
```

```
PROC BirdPos
    (BYTE xpos,ypos,char1,char2)
;Puts Two bytes, char1 & char2
```

```
;AT xpos, ypos on screen
```

```
CARD scmem=88
```

```
ac=scmem+xpos+(ypos*40)  
Poke(ac,char1)  
Poke(ac+1,char2)  
RETURN
```

```
PROC Song()
```

```
FOR a=0 TO 7 DO ;eight notes in song  
  b=notes(a)  
  c=dur(a)  
  d=10  
  e=notes1(a)  
  FOR ac=1 TO c*40 DO  
    IF ac MOD 100=0 THEN  
      d=-1 ;decrement volume  
    FI  
    Sound(0,b,10,d)  
    Sound(1,e,10,d)  
  OD  
  Sound(0,0,0,0)  
  Sound(1,0,0,0)  
OD  
RETURN
```

```
PROC Init()
```

```
;Initialize chset,pmg & playfield
```

```
Poke(106,memory) ;reset top of memory  
Graphics(0)  
Poke(559,0) ;turn ANTIC off  
;Display List  
ac=PeekC(560)  
FOR a=6 TO 24 DO  
  Poke(ac+a,4) ;IR Mode 4  
OD  
Poke(ac+25,164) ;DLI & VSCROLL on  
Poke(ac+26,164)  
Poke(ac+27,34) ;VSCROLL Set  
Poke(ac+28,34)  
;colors  
Poke(706,30)  
Poke(707,14)  
Poke(708,68)  
Poke(709,12)  
Poke(710,128)  
Poke(712,128)  
Poke(752,1) ;cursor off  
Poke(82,0) ;Left margin-0  
;Character Set  
a=Peek(106)-8  
chad=a  
Poke(106,a)  
Poke(756,a)
```

```

FOR ac=0 TO 1023 DO
  b=Peek(57344+ac)
  Poke(a*256+ac,b)
OD
MoveBlock(a*256+512,chset,224)
MoveBlock(a*256+776,chset2,160)
;Player missile graphics
a=-16
Poke(106,a)
Poke(54279,a)
Poke(53277,3)
Poke(623,52)
pmbase=a*256
Zero(pmbase,2048)
;Playfield
Position(14,0)
Print(".....")
;above is CTRL-Q R S T U V W X Y Z
Position(0,21)
Print(".....")
Print(".....")
;above is CTRL B B D 23-E's F B B
Position(0,22)
Print("  SCORE: 000000")
PrintE("          MEN: 00")
Print("  ENERGY: 00000")
Center(score,13,22)
Center(energy,14,23)
Position(31,22)
Print("0")
PrintC(men)
;DLI's
dlilvec=Dli1
vdslst=Dli1
Poke(54286,192)
Poke(559,62)
FOR e=0 TO 19 DO ;reset x & y values
  coordstore(e)=0 OD
FOR e=20 TO 29 DO ;random wing flaps
  coordstore(e)=Rand(2) OD
fallflag=0 ;disable meteor
RETURN

```

```

PROC CntFire()

```

```

;Continue firing
cc=PeekC(88)
bc=fy*40+fx
Sound(0,fy+fy+180,10,fy/2)
Poke(cc+bc,0) ;Erase Fireball
;Check for Illegal coordinates
IF fx=2 OR fx=37 OR fy=2 THEN
  fireflag=0
  Sound(0,0,0,0)
  RETURN

```

```

FI
;Increment positions
fx==+df

```

```

fy== -1
cc=PeekC(88)
bc=fy*40+fx
c=Peek(cc+bc) ;Object under fireball
Poke(cc+bc,219) ;fireball character
Delay(300)
IF c THEN ;check what under fireball
  FOR e=0 TO 5 DO ;Which bird hit?
    IF bexist(e)=1 THEN
      a=coordstore(e)
      b=coordstore(10+e)
      IF a<fx+2 AND a>fx-2 AND fy=b
        THEN
          bexist(e)=0
          BirdPos(a,b,0,0)
          PMove(pmbase,ball2,3,fx*4,
            fy*8,8);put explosion
          Delay(200)
        FI
      FI
    OD
  Sound(0,150,8,10)
  Delay(3000)
  ;Clear player 3 area
  Zero(pmbase+fy*8+1824,8)
  energy==+2
  fireflag=0
  Sound(0,0,0,0)
  score==+1 ;increase score
  Poke(cc+bc,0)
FI
Poke(cc+bc,0)
RETURN

```

```
PROC Title()
```

```

;Prints out title page
Graphics(17)
Poke(559,0);turn ANTIC off
;Display list
ac=PeekC(560)
Poke(ac+13,7)
Poke(ac+15,4)
Poke(ac+13,7)
Poke(756,chad+2)
Position(3,2)
PrintD(6,"ABEFABIJMNQR")
Position(3,3)
PrintD(6,"CDGHCDKLOPST")
Position(5,5)
PrintD(6,"PRESENTS")
Position(4,8)
PrintD(6,"12345 6789:")
Position(3,15)
PrintD(6,"BY DAN BULLOK")
Position(0,18)
PrintD(6,"  press start")
Position(5,10)

```



```

PrintD(6," .. .. .. ")
PrintD(6,".. ")
;above=space INVERSE CTRL-I J 4spaces
;CTRL-K L 4spaces CTRL-I J 4spaces
;CTRL-K L 4spaces CTRL-I J 2spaces
;PMG stuff
Poke(53277,3)
Poke(623,32)
Poke(704,28)
Poke(705,128)
Poke(708,12)
Poke(709,92)
Poke(712,134)
PMove(pmbase,p0,0,119,131,20)
PMove(pmbase,p1,1,119,131,20)
Poke(559,62);Turn ANTIC back on
WHILE consol#6 DO
    colpf3=random ;flash start
    wsync=0 ;wait for sync
    ;scroll colors in Demon Birds
    colpf2=128-vcount+(rtclock RSH 3)
    IF vcount=34 THEN
        chbase=chad
        colpf0=26
    ELSEIF vcount=41 THEN
        chbase=chad+2
    ELSEIF vcount=58 THEN
        chbase=chad
        colpf0=68
    ELSEIF vcount=65 THEN
        colpf0=168
    FI
OD
RETURN

```

```

PROC GameOver()

```

```

;Game Over message
SndRst()
gflag=1
Poke(106,memory)
Poke(623,4)
Poke(53277,0)
Graphics(17)
Poke(559,0)
Poke(708,14)
Poke(709,70)
Poke(710,128)
Poke(711,0)
Poke(712,136)
ac=PeekC(560)
Poke(ac+9,7) ;Graphics(2) at line 4
Position(5,4)
PrintDE(6,"game over")
Position(4,10)
PrintDE(6,"final SCORE:")
Position(7,12)
PrintD(6,"000000")

```

```

Center(score,10,12)
Position(4,18)
PrintDE(6,"press start")
Poke(559,34)
WHILE consol#6 DO
    wsync=0
    colpf3=vcount+rtclock/2
OD
RETURN

```

```

PROC NewMan()

```

```

;Materialize New Wizard
Zero(pmbase,2048)
Poke(704,78) Poke(705,78)
FOR a=0 TO 100 STEP 2 DO
    FOR b=0 TO 7 DO
        ball2(b)=ball(b)&random
        Sound(1,a+a,8,a/10)
    OD
    PMove(pmbase,ball2,0,a,y,8)
    PMove(pmbase,ball2,1,200-a,y,8)
OD
Zero(pmbase,2048) ;clear pm area
b=10
;materialize man
FOR a=0 TO 20 STEP 2 DO
    b=10-a/2
    PMove(pmbase,p0+b,0,100,y+b,a)
    PMove(pmbase,p1+b,1,100,y+b,a)
    Poke(704,30-a/10)
    Poke(705,140-a/2)
    FOR c=0 TO 100+a*6 DO
        d=255-c
        Sound(1,d,10,10-a/2)
    OD
OD
Sound(1,0,0,0)
Poke(704,28)
Poke(705,130)
x=100
y=154
fireflag=0
rtclock=0
RETURN

```

```

PROC Die()
;Death of wizard
;Puts player data in missile area
;and blows player apart into 4 pieces

```

```

BYTE ARRAY image(20)

```

```

Poke(704,14)
Poke(705,14)
;spins player around
FOR a=0 TO 15 DO

```

```

    PMove(pmbase,p0+40,0,x,y,20)
    PMove(pmbase,p1+40,1,x,y,20)
    Delay(1000)
    PMove(pmbase,p0+120,0,x,y,20)
    PMove(pmbase,p1+120,1,x,y,20)
    Delay(1000-a*30)
    Sound(0,155-a*10,10,a)
OD
SndRst()
Zero(pmbase,2048)
FOR a=0 TO 20 DO
    image(a)=p0(a)%p1(a) OD
FOR a=0 TO 20 DO
    image(a)=image(a) RSH 1 OD
MoveBlock(pmbase+800+y,image,20)
Poke(711,14)
;blows player apart
FOR a=0 TO 100 DO
    Poke(53254,x-a+48)
    Poke(53253,x-a/2+48)
    Poke(53252,x+a/2+48)
    Poke(53255,x+a+48)
    Sound(0,a/3,8,a/12)
    Delay(a)
OD
SndRst()
RETURN

PROC Move()

;move wizard
ctr==+20 ;image counter
IF ctr=80 THEN
    ctr=0 ;reset counter if too big
FI
x=x+DeltaX()
IF x<10 THEN x=10
ELSEIF x>142 THEN x=142 FI
IF DeltaX()=0 THEN
    ctr== -20 ;if player is not moving
    Delay(250)
    IF ctr>60 THEN ctr=60 FI
    ;If player stood still too long,
    ;Make him sink in the mud
    IF rtclock>80 THEN
        Birdpos(x/4-1,21,0,0)
        Birdpos(x/4+1,21,0,0)
        SndRst()
        FOR c=0 TO 24 DO
            PMove(pmbase,p0,0,x,y+c,26-c)
            PMove(pmbase,p1,1,x,y+c,26-c)
            Delay(3000)
            Sound(0,c+150,10,5)
        OD
        Sound(0,0,0,0)
        dieflag=1
    FI
ELSE

```

```

    Poke(20,0)
    PMove(pmbase,p0+ctr+dir,0,x,y,20)
    PMove(pmbase,p1+ctr+dir,1,x,y,20)
FI
IF ctr=40 AND DeltaX()#0 THEN
    ;click feet
    Poke(53279,0)
    Poke(53279,8)
ELSE
    Delay(250)
FI
IF fireflag THEN
    CntFire()
ELSEIF STrig(0)=0 THEN
    fireflag=1
    fx=x/4+1
    fy=20
    df=DeltaX()
    energy== -1
ELSE
    Delay(300)
FI
RETURN

```

```

PROC GetReady()

```

```

Graphics(18)
Position(5,5)
PrintD(6,"GET ready")
Poke(623,4) ;players behind playfields
Poke(53277,0)
FOR ac=1 TO 20000 DO
    wsync=0
    colpf0=128-vcount+rtclock RSH 2
    colpf1=vcount+rtclock RSH 2
OD
RETURN

```

```

PROC MainLoop()
    BYTE mcount,lum

```

```

;Infinite Loop
DO
    ;7 player moves to one bird move
    FOR mcount=1 TO 7 DO
        IF random<10 AND fallflag=0 THEN
            fallx=Rand(140)+10 ;drop meteor
            fally=10
            fallflag=1
        ELSEIF fallflag THEN
            fally==+5
            fallx==+Rand(5)-2
            FOR b=0 TO 7 DO ;random ball
                ball2(b)=ball(b)&random OD
            PMove(pmbase,ball2,2,fallx,
                fally,8)
            Sound(0,fally,8,fally/10)
        
```

```

IF fally>170 THEN ;hit bottom?
    fallflag=0
    Zero(pmbase+1536,256)
    Sound(0,0,0,0)
FI
FI
Poke(53278,1) ;hitclr
Move()
Poke(711,random) ;flash bird eyes
;kill wizard
IF energy=65535 OR Peek(53252)=1
    OR dieflag#0 OR Peek(53262)#0
    THEN
    men=-1
    energy=20
    SndRst()
    ;Turn birds off
    FOR e=0 TO 5 DO
        bexist(e)=0
        BirdPos(coordstore(e),
                coordstore(e+10),0,0)
    OD
    IF men=0 OR men>10 THEN
        gflag=0
        EXIT
    ELSE
        IF dieflag THEN
            dieflag=0
        ELSE
            Die()
        FI
        rtclock=0
        GetReady()
        Init()
        Newman()
        Poke(20,0)
    FI
FI
OD
IF gflag=0 THEN
    EXIT
FI
;Shake earth
e=Rand(4)
Poke(54277,e)
b=Rand(10)
Sound(1,50+b*20,8,e+3)
y=154-e
PMove(pmbase,p0+ctr+dir,0,x,y,20)
PMove(pmbase,p1+ctr+dir,1,x,y,20)
;If a bird isn't on screen,
;put it there if random<30
FOR e=0 TO 5 DO
    IF bexist(e)=0 AND random<30 THEN
        bexist(e)=1
        IF e MOD 2=0 THEN
            coordstore(e)=0
        ELSE
            coordstore(e)=39
        FI
    FI

```

```

    FI
  FI
OD
;Center score and energy
Center(score,13,22)
Center(energy,14,23)
Position(31,22)
Print("0")
PrintC(men)
;Start Key ends the game
;Option Key stops the program
;Any key pauses game
IF consol=6 THEN
  EXIT
ELSEIF consol=3 THEN
  Poke(106,memory)
  Graphics(0)
  Break()
ELSEIF ch#255 THEN
  ch=255
  WHILE ch=255
    DO OD
    ch=255
    rtclock=0
  FI
;Move all 6 birds
FOR bcount=0 TO 5 DO
  bx=coordstore(bcount)
  by=coordstore(10+bcount)
  BirdPos(bx,by,0,0)
  IF bexist(bcount)=1 THEN
    bflap=coordstore(20+bcount)
    char1=201+bflap+bflap+4*
      (bcount MOD 2)
    char2=char1+1
    bflap=flapinc(bflap)
    coordstore(20+bcount)=bflap
    bx==+increase(bcount MOD 2)-1
    IF bx=40 THEN
      bx=0
    FI
    IF bx=255 THEN
      bx=39
    FI
    coordstore(bcount)=bx
    by=strafey(bcount*40+bx)
    by=by
    coordstore(10+bcount)=by
    BirdPos(bx,by,char1,char2)
  FI
OD
OD
RETURN

PROC Game()

memory=Peek(106) ;Get top of memory
DO

```

```
;reset variables
Men=4
Score=0
Energy=50
Init()
Title() ;Title screen
Init()
Song()
Newman()
Mainloop()
;play song when game is over
Graphics(17)
Poke(712,134)
Poke(623,4)
Poke(53277,0)
Song()
GameOver()
OD
RETURN
```

[analog_28_demonbirds.pdf](#)