

GALACTOR

Galactor is a game written for the MSX platform in just 10 lines of BASIC code.

- `Title...` : Galactor
- `Platform` : MSX
- `Author..` : Martin Rizzo
- `Language` : MSX BASIC 1.0
- `Category` : PUR-120

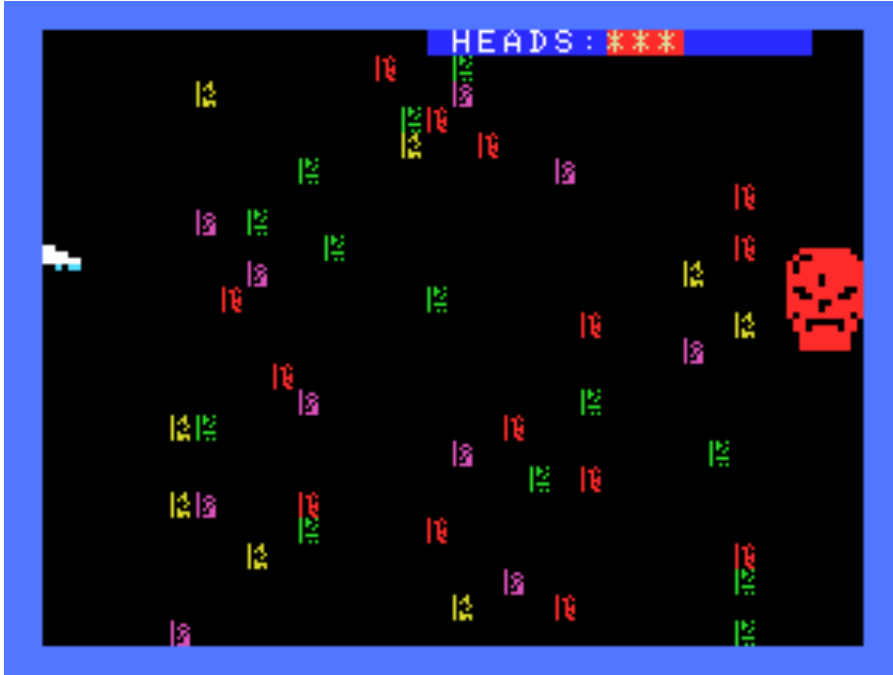
Files

GALACTOR.BAS	The BASIC source code of the game (MSX)
GALACTOR.GIF	Screenshot of the game
GALACTOR.DSK	720Kb disk image to use with the emulator
LIST-01.PNG	Screenshot displaying lines from 1 to 5
LIST-02.PNG	Screenshot displaying lines from 6 to 10
README.PDF	Full documentation in PDF
README.HTM	Full documentation in HTML
README.MD	Full documentation in Markdown format
README.TXT	Text file with a brief description of the game

Gameplay

You have to destroy all the 7 Galactor's heads to free the universe from this giant evil. On board your spaceship, you must fire and guide your missiles avoiding the energy barriers that protect him. Your time is limited because once Galactor gathers forces then he will go for you!.

- Use the [UP] and [DOWN] arrow keys to control your spaceship.
- Press the [SPACEBAR] to launch the missile.
- The missile can be controlled up and down to avoid the energy barriers.
- The game is over when Galactor gather enough strength to attack you.
- Destroy the 7 Galactor's heads and you win!



How to Run the Game

Loading the game in blueMSX emulator

1. Download blueMSX FULL v2.8.2 (or above) from some of these links:
 - [blueMSX home page](#)
 - [blueMSX download page](#)
2. Install the emulator following the setup wizard.
3. Start blueMSX
4. Select "File > Disk Drive A > Insert"
5. Select the provided .DSK file (the disk image)
6. Select "Emulation > Run" and enjoy the game!

Source Code

```

LIST 1-5
1 DEFINT A-Z:K=1024:Q=32:SCREEN1,3,0:WIDT
H0:COLOR15,1,5:KEYOFF:ONSPRITE60SUB10:J(
1)=-1:J(5)=1:T=RND(-TIME):PLAY"S14M63L64

2 FORI=0TO11:U=I\2:READT,M$(U):SPRITE$(U
)=M$(U):VPOKE8196+I,T:FORJ=0TO31:VPOKE(I
+12)*64+J,RND(1)*256AND&H3COR128:NEXTJ,I

3 H=6:F=232:Y=9:LOCATE10,9:PRINTM$(5):IF
STRIG(0)THENFORJ=9TO12:POKE&HF923,J*4:CL
S:PRINT," 1HEADS:*****1 "":NEXTELSE3

4 FORI=0TO9:E=(E+3)MOD23:T=37+Q*E+24*RND
(1):FORJ=9TO12:VPOKEJ*K+T,87+8*(3ANDI)+J
:NEXTJ,I:X=0:G=192:D=6-H\2:M=60+11*H:C=1

5 P=(PAND3)+9:VDP(2)=P:M=M-1:Y=Y+J(STICK
(0))-(<Y<=3)+(<Y>=21):IFM>0THENX=X-(<X>0)AN
DX<29:G=G-D:IF0<GANDG<160RG<-149THENG=-G

Ok
█

```

```

LIST 6-10
6 S=S+(S>32):PUTSPRITE4,(X*8ANDM>0,Y*8):
7:PUTSPRITE2,(F,ABS(G)),8+M\25,2:PUTSPRI
TE3,(0,Y*8-6):IFSTRIG(0)THENX=1:PLAY"BC"

7 VPOKEP*K+R,S:T=Y*32+X+1:IFVPEEK(P*K+T)
>95THENR=T:S=34:X=0:PLAY"N58"ELSEIFM<0TH
ENF=F+12*(F>7):G=ABS(G+8+16*(G>8*(Y-2)))

8 SPRITEON:IFCTHEN5:DATA81,0,184,"GAME10
VER",68,0,244,"<YOU1WIN>",244,0,244,"?08
8TJf)*T8'o???_a=====e0==_<_",244,0,244

9 PLAY"CDEN33":FORJ=9TO12:VPOKEJ*K+22+H,
48:NEXT:IFM>0ANDH>0THENH=H-1:GOTO4ELSEVD
P(2)=12:LOCATE11,8:PRINTM$(-<H>0):GOTO3

10 SPRITEOFF:PUTSPRITE2,(F,ABS(G)+8),8,0
:C=0:RETURN:DATA"_"=,"33,0,&H81,","&H41
,,&Hd1,"PRESS1SPACE":'by Martin R. Rizzo

Ok
█

```

Some Code References

Variables & constants

```

<0,Y> = SHIP POSITION (in chars)
<X,Y> = MISSILE POSITION (in chars)
<F,G> = GALACTOR'S HEAD POSITION (in pixels)
J(n) = SHIP MOVEMENT INDEXED BY JOYSTICK DIR
H = REMAINING GALACTOR'S HEADS
D = DIFFICULTY (head movement velocity)
E = NEXT ENERGY BARRIER Y-COORD (in chars)
T,U = TEMPORARY VAR
I,J = LOOP VARS
M = COUNTDOWN TIMER
C = COLLISION (0=YES, 1=NO)
K = CONSTANT =&H400 =1024
P = CURRENT VIDEO PAGE [9,10,11,12] ' &H2400, &H2800, &H2C00, &H3000
R = LAST REMOVED ENERGY BARRIER (ADDRESS)
S = LAST REMOVED ENERGY BARRIER (CHARACTER)

```

Lines

```

1-2 : initialization
3 : start new game
4 : start next level
5-8 : game loop
9 : headshot or game-over
10 : sprite collission detected

```

