

LPrint Runaway

by @ignacobo

<http://ignacobo.itch.io>

The original LCD Card, game&watch style, was developed in 1983 by the Japanese firm Gakken.



Instructions read:

“Runaway from the prisoners’ camp. Advance to the fence without being spotted by search light and not discovered by warders. When 3 prisoners are discovered or spotted 3 times, the game is over. “

This is the conversion to Pure Basic for ZX Spectrum 48kb, around 40 years later.

Contestant at Bytemaniacos Basic2023 by Radastan.



Special thanks to:

IvanBasic (@_IvanBasic_) for explaining this LPrint technique, which he used thoroughly in Brain8 game.

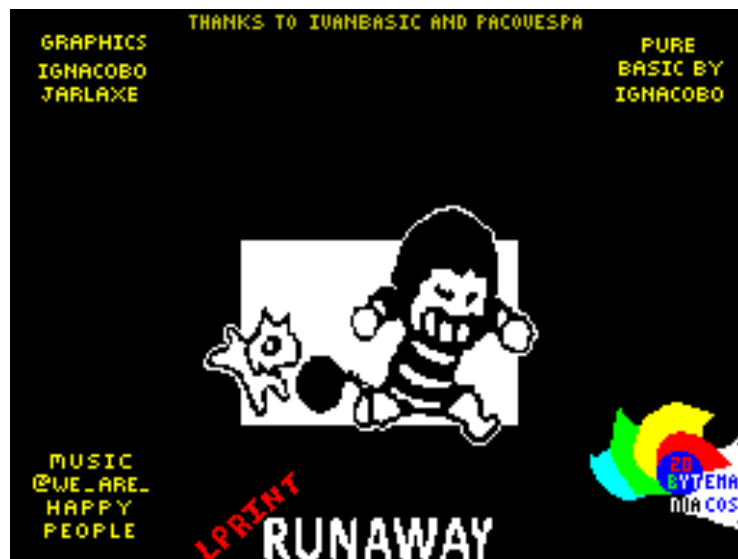
Jarlaxe (@JarlaxeRetroGa1) for guiding me through all graphical part of the game. Sprites in Runaway bear his seal.

@we_are_happypeople for composing the catchy tune

Pacovespa (@Pacofranquelo) for betatesting with care and love for detail.

AnbJørn Myren for his advice on the original game design and mechanics.

Radastan (@Bytemaniacos), for his restless support to developing projects in Basic, and Retro in general, through his programming jams.



Keys:

Control your prisoner with

O to move to the left,

P to move to the right.

M to toggle music on and off

Technical info:

Technique uses LPrint on attribute zone to rapidly change colors on the screen. The bitmap remains unaltered all the time. The code has only one Print instruction, used to update the score. All movement sensations are achieved with fast change of ink and paper color, directly on attribute memory, using the technique detailed on this blog:

http://blog.jafma.net/2020/08/29/efficient-basic-coding-for-the-zx-spectrum-v/#en_5

The toughest part of the design was developing each 8byte group with ink and paper values necessary at every spot of the screen. I used 21 UDG and a full charset (21+96 chars!) to get all the possibilities I needed.

It has in-game music, playing one beep note every 16 frames.

Speed and difficulty increase every 10 seconds and as score gets higher and higher.

Code:

To start, execute RUN 9000 command, where variables are set. Auxiliary files are already loaded into memory from Intro.bas program.

After a BREAK, type to read the listing.

I use DefAdd to rapidly draw colors with initial screen (768bytes) every time you lose a life.

Main loop, from lines 0 to 20,

[0] print score. This is the only Print instruction in the code!

[4] jumps to drawSearchLight, drawLeftWarder or drawRightWarder depending on time T.

[6] Plays a beep note. All Music logic in one line!

[8] Speed gets faster every 10 seconds, and also as score gets high.

[10-11] Reads keyboard and jumps to movePrisonerLeft or movePrisonerRight if required

All these subroutines are composed of many Pokes setting position on the attributes screen and some LPrint to show Colors.

[30-60] movePrisonerRight

[66-96] movePrisonerLeft

[110-180] drawSearchLight

[200-241] drawRightWarder

[300-361] drawLeftWarder

[1200-1350] drawPrisonerAtFence

[6000-6100] lose one life, draw prisoners in jail, check Game Over and set HighScore.

[9000-9100] Start of program, variables initialization and new Game.

Files:

DefAdd, 4 variable data to move colors to attribute zone in a fast way

Nums64cx, 4pixel wide numbers to draw the score

ColOscuro, intro color scheme to get a nice effect if loading from tape

Music y durMusic, music notes and duration

Gak.bin, Intro screen bitmap

Timing, data pattern to move the searchLight (1cx), the leftWarder(3x) and the rightWarder(4x)

ColoresIni, starting game color scheme.

lcdInv4.bin, main game screen bitmap

Charset y UDG, ink and paper patterns needed to draw all characters , 96+21!!

RunAway.bas, game's code in pure Basic
