|  |
| --- |
| The Trashcan (Software and Media Publication, NFP) |
| The Dinosaur Game |
| Computer Game Design Document and Development Log |

|  |
| --- |
| Ray Arias  9 February 2018 |

Contents

[I. History 4](#_Toc506460413)

[A. Original 1986 Game Look 4](#_Toc506460414)

[B. Original Game Music 5](#_Toc506460415)

[C. Original Game Play 5](#_Toc506460416)

[1. Description in Prose 5](#_Toc506460417)

[2. Pseudocode 6](#_Toc506460418)

1. History
2. Original 1986 Game Look

The current design of this game is based on a simple game that Ray made in 1985-6 on a Commodore 128 in Microsoft Commodore BASIC 7.0 and 8502 machine code using the built in machine language monitor. There are absolutely neither any soft- nor hardcopies of any code, screens, or design for this original game. The only source for information about this game is Ray’s memory of it. Any reference to this game in this document will be labeled TOG for “The Original Game.”

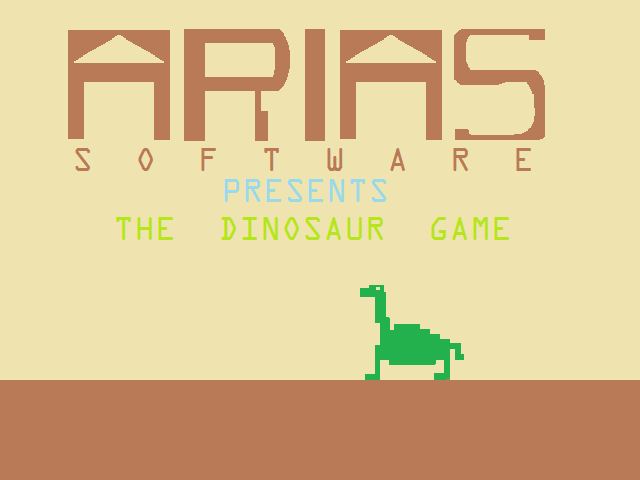
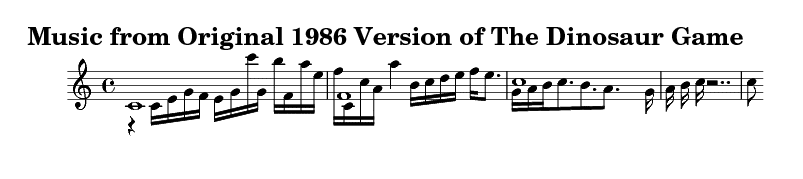
TOG had a start screen with a dark brown ground comprising about the bottom fifth of the screen, a yellow background making up the remainder of the screen and a green dinosaur with a Brontosaurus/Apatosaurus appearance walking across atop the ground from right to left with deep percussive tones played with every step. The size of this dinosaur was approximately a fifth of the screen vertically and a seventh of the screen horizontally, and it was physically about as tall from head to foot as it was long from head to tail. Displayed at about mid-height of the start screen were large letters saying “ARIAS” and under these were medium letters spaced to fit the horizontal space taken up by the large letters saying “S O F T W A R E” and under this were more medium letter saying “Presents THE DINOSAUR GAME.”

Figure 1: Initial screen of original 1986 version of The Dinosaur Game, recreated from memory by Ray.

1. Original Game Music

The music for TOG was played with the three-voice VIC sound chip in two of its three voices. The lower voice was an organ and the higher voice was a guitar. 

1. Original Game Play
2. Description in Prose

TOG began when the fire button on the joystick was pressed and displayed a dark blue screen with a brown ground bottom in the same position it was in the start screen. The player was displayed on the left side of the screen as a white human-looking figure with what appeared to be a spear in its hand. In the bottom, lettering was displayed as a message line by line to the player explaining the game story (see next subsection, D. Original Game Story), until either the fire button was pressed again or 10 seconds passed after the final line of text. Then a long tone ascending in frequency would be played and, when the tone rose to top pitch, the same yellow background with a brown bottom as the start screen would be displayed, and another descending tone starting at high frequency would be played. When the tone lowered to its bottom pitch, the dinosaur would appear all the way on the right edge of the screen and play would begin at 3 lives and first level.

When the joystick was left alone, the player figure would stay still and the dinosaur figure would proceed left toward the player at a certain speed. (At Level 1, this speed was about 5 pixels a second, but this speed would increase at higher levels.)

When the joystick was pushed left, the player figure went about 20 pixels to the left, up to the left edge of the screen.

When the joystick was pushed right, the player figure went about 20 pixels to the right, theoretically up to the right edge of the screen, although because the dinosaur was inevitably in the way, this typically was not possible. (Sometimes, a glitch early on in the development of TOG would actually make this possible because the routine that detected whether the player and dinosaur collided would not run fast enough—you gotta love Commodores running BASIC—before the player passed to the other side of the dinosaur. However, this issue was later solved by matching the X position of both the player and the dinosaur in such a way that if the X position of the player was greater than or equal to the X position of the dinosaur, the routine that causes the player to die would execute. However, this method of detection was used concurrently with the traditional collision detection method, as opposed to in place of it.)

Pushing the joystick up or down, while early on in development of TOG had no effect, later had the effect of allowing the dinosaur to proceed somewhat faster during the time the joystick was pushed either up or down. When the joystick was released from the up or down position, the dinosaur would again proceed at its original speed.

When the fire button was pressed, a spear coming from the player’s position would be released and travel at 45° counterclockwise from the horizontal until it got to the top of the screen, at which point the spear would appear to turn 135° clockwise, such that it proceeded straight downward. If the spear did not hit the dinosaur, play would continue as before. However, if the spear did hot the dinosaur, the dinosaur would vanish, the player would be advanced to the right until the player figure wrapped around to the left side again (in what was presumably another scene, but appeared to be the same) and would stop at the same position it was at the beginning of the first level, another dinosaur would appear at the right edge of the screen, and play would start again at a new level.

1. Pseudocode

void DisplayStartScreen() {

DisplayTitleAndBackground;

PlayThemeMusic;

int XPositionDinosaur = RightEdge;

int DinosaurFigure = 1; /\* There are 4 dinosaur figures according to what position the Dinosaur’s feet are in \*/

while (FireButtonNotPressed) {

while (DinosaurFigure <= 4) {

DisplayDinosaur(DinosaurFigure, XPositionDinosaur);

DinosaurFigure++;

XPositionDinosaur -= 5;

if (FireButtonNotPressed)

XPositionDinosaur = RightEdge; }

Dinosaurfigure = 1; }

StopThemeMusic; }

void GamePlay() {

DisplayStoryBackground;

While (FireButtonNotPressed)

for (int StoryLine = 0; StoryLine <= LastLineOfStory; StoryLine++)

DisplayStoryLine(StoryLine);

DisplayStandardGameBackground;

int Level = 0;

int Score = 0;

int Lives = 3;

int XPositionPlayer = LeftEdge;

int XPositionDinosaur = RightEdge;

int DinosaurFigure = 1;

int PlayerFigure = 2; /\* There are 3 player figures according to if the player is standing still (PlayerFigure = 2), moving to the left

(PlayerFigure = 1), or moving to the right (PlayerFigure = 3). \*/

DisplayPlayer(XPositionPlayer, PlayerFigure);

while ((XPositionPlayer < XPositionDinosaur) && (DinosaurAlive)) {

if (NoJoystickMovement) || (JoystickUp) || (JoystickDown)) {

if (DinosaurFigure <= 4) {

DisplayDinosaur(XPositionDinosaur, DinosaurFigure);

DinosaurFigure++;

XPositionDinosaur -= 5;

if (NoJoystickMovement) Wait(Level);

else if (Level < (MaxLevel – 2))

Wait(Level + 2);

else Wait(MaxLevel); }

else DinosaurFigure = 1; }

else if ((JoystickLeft) || (JoystickDiagUpLeft) || (JoystickDiagDownLeft)) {

int InitPlayerPosition = XPlayerPosition;

int FinalPlayerPosition = XPlayerPosition – 5;

if (FinalPlayerPosition < LeftEdge)

FinalPlayerPosition = LeftEdge;

PlayerFigure = 1; /\* Facing left \*/

while (XPositionPlayer > FinalPlayerPosition) {

XPositionPlayer--;

DisplayPlayer(PlayerFigure, XPlayPOsition); }

}

if ((JoystickRight) || (JoystickDiagUpRight) || (JoystickDiagDownRight)) {

int InitPlayerPosition = XPlayerPosition;

int FinalPlayerPosition = XPlayerPosition + 5;

PlayerFigure = 3; /\* Facing right \*/

while (XPositionPlayer < FinalPlayerPosition) {

XPositionPlayer++;

DisplayPlayer(PlayerFigure, XPlayPOsition); }

}

else if (FireButtonPressed) {

int YPositionSpear = Ground;

int XPositionSpear = XPositionPlayer;

int AngleSpear = 45;

bool DinoHit = FALSE;

bool SpearFigure = FALSE; /\* There are 2 spear figures, one for shooting up at 45 deg angle and one for going straight down \*/

for (int YPositionSpear = Ground;

YPositionSprear > TopOfScreen;

YPositionSpear -= sin(AngleSpear) {

XPositionSpear += cos(AngleSpear);

DisplaySpear(SpearFigure,

XPositionSpear, YPositionSpear); }

AngleSpear = 270;

SpearFigure = TRUE; /\* Switch to falling spear \*/

For (int YPositionSpear = TopOfScreen; YPositionSpear < Ground; YPositionSpear++)

DisplaySpear(SpearFigure,

XPositionSpear, YPositionSpear);

DinoHit =

(Abs(XPositionSpear – XpositionDinosaur) <= 7);

If (DinoHit) {

KillDino;

Score += 100;

Level++; }

}