Komoda & Amiga plus - Commodore computer user magazine

ISSN: 2956-4735

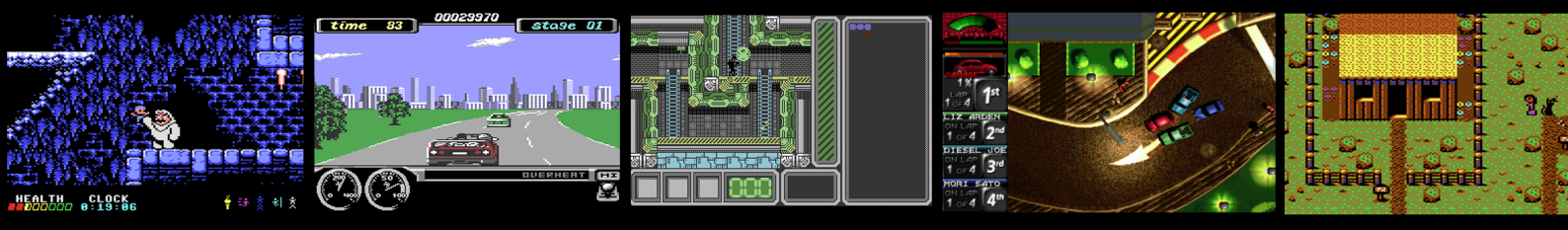
K&A plus #25

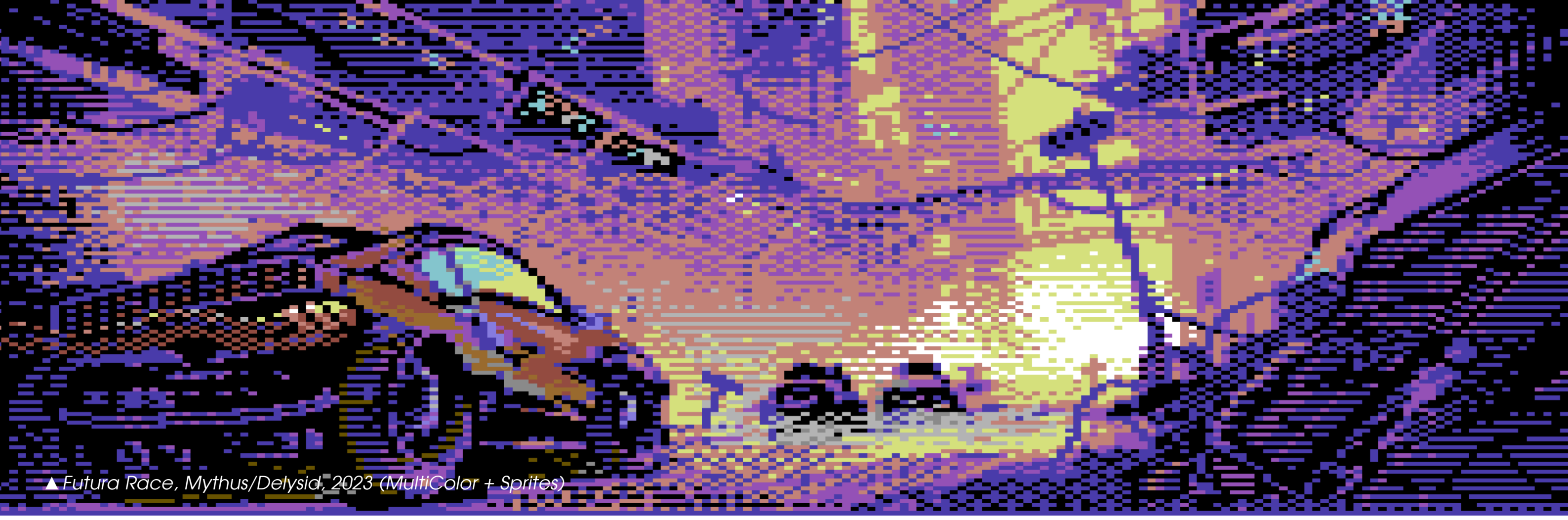
spring/summer 2024

English edition

Lotus 1-2-3
3D Rat Race
A view into racing

Party Speedway Extd. GameDev Story
Interviews: Friedrich Volmering, Lasse Öörni, Paolo Cattani
New releases: Yeti Mountain, Turbo Outrun, Harharagon




▲ Futura Race, Mythus/Delysid, 2023 (MultiColor + Sprites)

Intro: Let's race!

I think everyone was passionate about cars, or 'racing' in the kids' jargon, when they were growing up. At first it was toy cars, real or fake Matchboxes, depending on the wealth of your parents' wallet, although I don't remember the latter. What I do remember is a miniature racetrack that took up half my room. My brother and I would race our 'formula cars', accelerating with a one-button plastic joystick. During the holidays we would take to the yards and pavements of the estate. We raced everywhere, each of us putting on the track the best cars from our collection.

Soon, we were all sitting in front of our blocks playing the electronic handheld racing games. We broke records, fine-tuned tactics and discovered possibilities that the creators of these machines had never dreamed of. Each game was fun because what we couldn't see on the small liquid crystal screen was filled in by our imagination. Back then, we could enjoy simple things. And that was the best.

When the first 8-bit machines arrived in our homes, racing moved back into the flats. Even if we weren't particularly moved by cars, we all felt its powerful touch as we fired up a compilation of racing games and sat with a mate's C64 with five other people. At first it was simple racing with simple rules: accelerate, brake, two gears - slow and fast, avoid opponents - no big deal. Later there were additional obstacles, oil slicks on the road, skidding, the ability to choose automatic or manual transmission. That was something. But it wasn't until the next element that we couldn't take our eyes off the screen or our hands off the joystick - split-screen racing. We could race against a buddy, laughing when we overtook him and cursing when we fell behind. Two players, two cars not controlled by the computer, all on one screen - it was something amazing.

The games kept changing and we kept discovering new titles and new possibilities. For example, we could choose tyres and change them at the pit stop. We could get off the track and watch little people change the wheels on our car. Not to mention adjusting the shock absorber stroke or tampering with the turbine. A simple crash, a burning engine, a cracked windscreen or a broken wheel was something many had been waiting for for years.

Later, games appeared in which the word 'race' was not always associated with sporting competition. When someone mounts a machine gun on the roof of their car, you start to doubt whether you are participating in a race or a desperate escape and fight for your life. But even this departure from the genre does not detract from the fun. Games are first and foremost entertainment.

In this issue, we're taking you back to the world of 8- and 16-bit racing, which is still alive and kicking after all these years. New games and conversions of old hits encourage us to get back behind the wheel of our dream car, taking us back to where our racing memories began. As long as our virtual car is equipped with a radio and CD player, we can open the roof of our dream convertible and play the music at full volume while enjoying the beautiful weather.

We hope you enjoy reading this issue and that it brings you many happy memories and unforgettable experiences.

Best wishes, Retrobajtel and the editorial team

We are committed to ensuring that the ratings of the games we review are fair and objective. Therefore, from this issue, we are changing the rules for awarding them. Score 5 will be the starting score. It will be awarded to games that are worth playing. Ratings above 5 will be reserved for titles that stand out in terms of playability and presentation (interesting graphics, music, original scenario).

CONTENTS

spring/summer 2024

COMMODORE

▪*Plus/4* ▪*VIC-20* ▪*C16* ▪*C64* ▪*C128*

AMIGA

▪*M68k* ▪*AOS 4.x* ▪*Aros* ▪*MOS*

VARIOUS

▪*news* ▪*editorials* ▪*reports* ▪*interviews* ▪*emulation* ▪*drawings*

COVER DISK

This time we have two cover disks for our readers - the first is the Party Speedway Extd game by Tomxx, and the second is the Windigo Productions game pack. Both disks are also available as collector's boxed editions on a floppy, cassette and cartridge, which you can read more about on the following pages.

Komoda & Amiga plus
Commodore computer user magazine

Editor-in-chief: Arkadiusz 'Komek' Kosiarski
Managing Editor: Tomasz 'Tomxx' Ankudowicz

Editors: Rafał 'Don Rafito' Pączko, Ariel 'Ari' Zimny, Marcin 'phowiec' Królikowski, Tomek 'tommysi' Garbarski, Paweł 'Monka' Marcinek, Agnieszka 'FPWG' Poziomka, Louie Dimovski, Łukasz 'Retro Chłop' Gniedziński, Tomasz 'Razor' Kaniecki, Erik Hooijmeijer, Maciej 'Void' Małecki, Tomasz 'Sleeva' Wiśniewski, Maciej 'Retrobajtel' Szmatloch, Tomasz 'Jackal' Jędrusiak, Marcin 'Tect' Wieczorek, Tomasz 'Beszcza' Beszczyński, Łukasz 'Bóbr 8bit' Bobrecki, Mariusz 'Ramos' Rozwadowski
Design & DTP: Tomxx, Monka, Razor

Publisher: *K&A Arkadiusz Kosiarski*
ul. Felińskiego 23/1A, 41-923 Bytom, Polska *NIP: PL6262591695*

Cover design: Monka
Editing and rewrite of Polish edition: Krzysztof 'Drakon' Odachowski, Marcin 'Piana' Pianka, Jacek 'Deadman' Miśkiewicz, Maciej Sznurkowski, Michał Sobczyk, Łukasz 'Hery' Herman
Translation: Ari, Tomxx, Piana, Monka, Toby, Void, Azu, Krzysztof Drzazga, Juan Calderon
Editing and rewrite of English edition: Martin 'Logiker' Roscher, Louie Dimovski, Erik Hooijmeijer, Wojciech Bagniewski
Contributors: Jan 'Johnny' Lorek, Leopold Tupalski, Krzysztof 'Drakon' Odachowski, Piotr 'c00k' Machnowski, Szymon 'netdemon' Trygar, Martin 'Logiker' Roscher

WWW: https://ka-plus.pl E-mail: team@ka-plus.pl Facebook: @KomodaAmigaPlus Twitter: @KAplusmagazine

Fresh News

C64

Hermanos Catrat

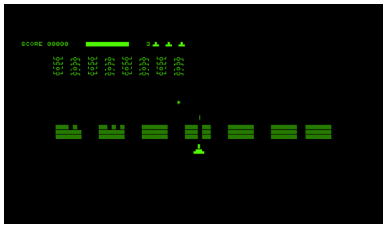
D. Rejment

The game revolves around the titular cat, who must overcome numerous obstacles to escape the depths of hell. The player has the ability to pass through flaming portals, which adds a strategic element to the gameplay. The gameplay in Hermanos Catrat is dynamic and challenging, ensuring that the game never gets boring. The levels are varied and offer a variety of challenges and puzzles to solve. In addition, the game contains many secret passages and hidden treasures, motivating the player to explore the map. Some of the game's stages may be too difficult for less experienced players, but for fans of retro platforming, Hermanos will be a real gem. ■ **Agnieszka FPWG**

Commodore 8032/SuperPET

Spyders

milasoft64

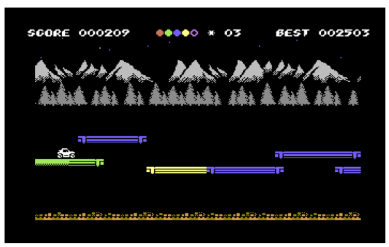


Space Invaders clone on PET. Instead of shooting aliens, you shoot spiders. Players with arachnophobia don't have to play it - unless they want to tame their fear. Others can see if they like this very simple implementation of a classic. Enemy on top, you on the bottom. The wheel reinvented, but nothing wrong with that. PET fans like games like this too. ■ **phowiec**

C64

Cruisin' For A Bruisin'

Megastyle



Buckle up and step on the gas! Our Bumblebee is ready for action! Crusin' For a Brusin' is a brilliant platformer about a jumping car. Our job is to drive over bridges, jump and pick up hitchhikers. We get points for all this. Simple mechanics, but lots of fun and brilliant music. Check it out and avoid the grey footbridges! ■ **Beszcza**

Amiga OCS/ECS

Jump 'n Bump

Earok

This is a small production focusing on multiplayer competition for up to four players. We move rabbits around and score points by jumping on the opponent's head. The one with the most points wins. Despite the nice graphics and cheerful music, this title has nothing more to offer (after all, it is only a beta version). In the full version we will get a single player mode and more boards. As it stands, this title offers a fun kicking contest. It's worth giving it a try with a second person. ■ **Tect**

Plus/4

The Great Giana Sisters

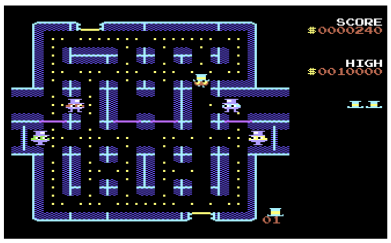
TCFS, Unreal, Csabo

Ladies and Gentlemen! Giana and her sister Maria invite you to another adventure, to relive it even, because a new conversion of The Great Giana Sisters for the Commodore Plus/4 has just been released. I started the game and... the nostalgia kicked in immediately. Ah! What a faithful conversion. Everything is as it was years ago (well, almost, because we know that the technical capabilities of the C64 and the Plus/4 are slightly different), the same unique atmosphere, only the Sisters are somehow slower and less powerful. Well, age does its work... Thanks to TCFS, Unreal and Csabo for the pleasant surprise and I'm off with my sisters, because we still have a few levels to go. ■ **Komek**

C64

Lock'n'Chase

LC-Games



It's time for a game of cops and robbers, but one from 1981. And it's not hard to guess that this is another conversion (this time from arcade machines) for the C64. In the game you control a thief who moves through a maze. He has to collect coins (dots) and treasures in it and at the same time not get caught by the cops. The game resembles Pac-Man in structure, guaranteeing a lot of fast and pleasant entertainment - as in most games from LC-Games. ■ **Komek**

Amiga

Ami H.E.R.O.

K&A+

Ami H.E.R.O. (Helicopter Emergency Rescue Operation) is an arcade game with a strong touch of puzzle logic. We take on the role of the protagonist of a one-man rescue team, armed with a laser blaster, a few sticks of dynamite to reach blocked corridors and a rotor-craft propulsion system to move vertically without worrying about damage to our health when we hit the ground after a long flight at altitude.

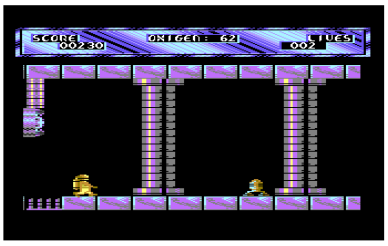
The player's task is seemingly trivial. Descend as low as possible into a cave (or mine) to rescue trapped miners (or cavers). That's the arcade part - we control our hero with the joystick, plant dynamite on walls that block access to lower corridors, shoot spiders, bats, rock snakes, poisonous moths and avoid natural traps. The logic part is choosing the right path from several possible ones and exercising your memory of what goes where in order to avoid or eliminate it. The difficulty increases quite rapidly from level to level, and as you get lower and lower it turns out that five lives is not such an exorbitant number after all. The graphics are pleasing to the eye (the protagonist looks a bit like the guy from Benefactor - and that's a compliment), the music in the intro is quite relaxing (as opposed to the music during the game, which increases the adrenaline rush and creates a mood of fear and danger), and the sound effects are very good, including some digitised screams.

As you know (or are about to find out), K&A Arkadiusz Kosiarski Publishing is not only responsible for the magazine you are holding in your hands, but also for the game releases. Ami H.E.R.O. will take up a 3.5 inch floppy disc and will be released in a boxed version on a physical medium. It's time to fire up the Amiga and clear DF0: in anticipation of production. ■ **Sleeva**

C64

Metal Piramid

Operation Mindcryme



We are still on the subject of Giana Sisters, because recently a game reminiscent of the adventures of our favourite sisters has seen the light of day (under the title Metal Pyramids). It should be noted that the game was created in the 1990s, so it has the unique atmosphere of that era. In particular, I am referring to the style of the graphics, which took me years back in time. So where does the game take us? To a space pyramid where we navigate a brave cosmonaut. We have to help him get a spaceship and the fuel to launch it. All in all, Metal Pyramid is a cool platformer with fairly typically designed levels (as befits this type of game). Play it and you'll see.
■ **Komek**

Amiga

Moon Patrol

JOTD

Oh yeah! I love Moon Patrol because it reminds me of childhood, carefree days and the local arcade. I used to play it a lot, so much so that my tongue was hanging out! And now we have the opportunity to play the arcade version on our beloved Amiga. As I write this, the game is already in advanced beta and is indeed a faithful arcade conversion. Just the same unique atmosphere. Same graphics, music and sound effects. What can I say, I have stepped back in time. I also have a funny anecdote about the game. Well, a few years ago I was playing Moon Patrol on an Atari 800XL with my friend and I noticed that the jumping and shooting Buggy in the Atari version looked like a "squashed anteater". And my friend said: "So what?" and he burst out laughing, he, he, he. ■ **Komek**

C64

Legend of Wilf

Hayesmaker64

Your name is Tony Wilf and you are the apprentice of Archmage Urlich. You will now travel 10,000,000 years BC to obtain all the parts of the Dragon Amulet. The Archmage casts a transformation spell on you and you grow wings. You look, er, like a moth? OK, no time to moan, every minute counts. You have a long way to go through a cave, a mansion, a castle... there will also be some diving for a change. You'll come face to face with dinosaurs,

bats, knights, sharks... it won't be easy. Overall the game is interesting and you can feel the atmosphere of most C64 productions from 1984/85. I like it! ■ **Komek**

Amiga

Glubble

Oxygene

Oh boy, I can't remember the last time I was so engrossed in a puzzle game. Glubbles are funny coloured noodles that want to connect (stick) to each other according to their colour. The noodles can only be moved in two directions (left and right) with the mouse. It may seem easy at first, but in the later stages of the game you will have to think hard. The game has pleasant graphics and soft music, and the noodles' funny and quirky faces will put you in a good mood. It's worth mentioning that Glubble is inspired by Jelly no Puzzle, which was released on PC and Android. So off you go, glue the noodles together. ■ **Komek**

VIC-20

VIC-20 Toybox

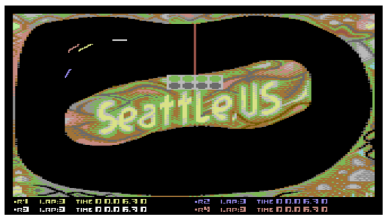
W.E. de Villiers

This toybox is a collection of 5 games, perhaps not quite finished but very playable nonetheless. There is Chilopoda, a centipede clone, Mars patrol featuring some sampled speech, a Petscii based breakout clone and of course Tetris, this time called Victris. Finally there is a train parking game. This is a good collection of games with some lovely graphics. Get it at huffelduff.itch.io and don't forget to read the footnotes in the readme.txt! ■ **Erik**

C64

Party Speedway Extended

K&A+



The Extended version is an extension of the pixelated speedway racer we released in 2022. The new version introduces 4 additional tracks, computer opponents on 4 difficulty levels and new modes: Tournament and Time Trial. The game will be published by our magazine in physical form on floppy disk and cartridge, and in traditional digital form as a d64 image. You can read about the process of creating the game and the programming puzzles involved in a special article on page 58. ■ **Tomxx**

C64

Kiełbasa Empire

Ordered Chaos Design

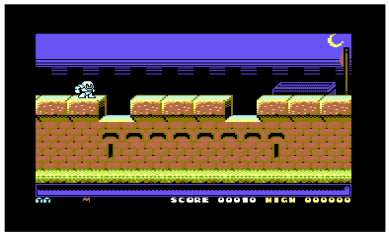
Kiełbasa Empire (ang. Sausage Empire) is a strategy game with adventure elements, in which we play the role of a young entrepreneur who has been thrown out of the state farm and must build a new sausage empire. To do this, you have to manage the sowing of the fields, the pigs and the food production. Between buildings, we move around a large map of the surrounding area. We can also go to other places, such as the shop or the platform, and indulge in other activities or mini-games. All of this was included in the latest demo of the game. There are many more ideas and I am keeping my fingers crossed that they will all come to fruition to complete this promising production.

You can follow the progress at https://ordered-chaos-design.itch.io/kielbasa-empire. Perhaps someone would like to support the project with their musical and graphic skills? ■ **Monka**

C64

Good Kniight

Icon64/Psytronik Software

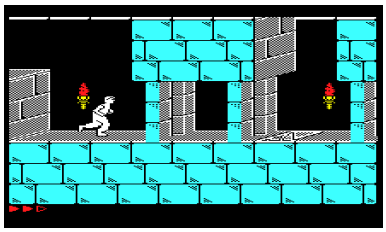


Icon64 pays tribute to Ocean's Hunchback with their release of Good Kniight. Set within a medieval theme, take control of a heroic knight as he crosses the battlements in order to rescue the princess from the evil dragon. Containing 40 plus screens, and a wide range of enemies and hazards, Good Kniight is a visually impressive platformer with tight controls and modern day game play sensibilities. The game is available now from Psytronik Software in both digital and physical formats and is sure to bring back many great memories of early 1980s C64 gaming. ■ **Louie**

VIC-20

Prince of Persia

Pedro Bermejo



Prince of Persia is in a race with Doom to see how many platforms it will be released on. Granted, no one has yet come up with the idea of porting Prince to a pregnancy test or a calculator, but who knows what the future holds. For now, 8-bit computers take precedence. The new winner in this category is the VIC-20. Pedro Bermejo, the author of the port, was inspired by the Spectrum version of Prince of Persia. Since this version uses 128 kB of RAM, Pedro had to make some hardware concessions. Still, to play the new old Prince, you need a 35 kB memory expansion and the PAL version of your computer (or an emulator). ■ **phowiec**

K&A+ Collector's Edition project

author and translation: Tomxx

EDITORIAL

As a publisher with a well-established position in the world of retro computing, we decided to expand our business profile by publishing boxed editions of C64 and Amiga games. We gained our first experience by publishing compilations of games on floppy disks for several years, cassettes and recently even cartridges under the GO8BG brand. With programmers, artists and design enthusiasts in our ranks, as well as extensive industry knowledge, we feel we can take the next step. We certainly won't be short of ideas or enthusiasm.

We are not targeting AAA games (at least for now), but we will do our best to ensure that the releases are rich in content, high in quality and elegant in appearance. We want to achieve all this while keeping prices affordable. That is why we have designed a universal foam packaging for multiple media: 3.5'' and 5.25'' floppy disks, a cassette tape and a C64 cartridge. Depending on the platform and media you choose, we will be able to provide you with a variety of productions in the formats of your choice. Admittedly, the box will not be made of hard cardboard (it will not be particularly rigid) as this would increase production costs, but we hope it will meet the expectations of retro game collectors.

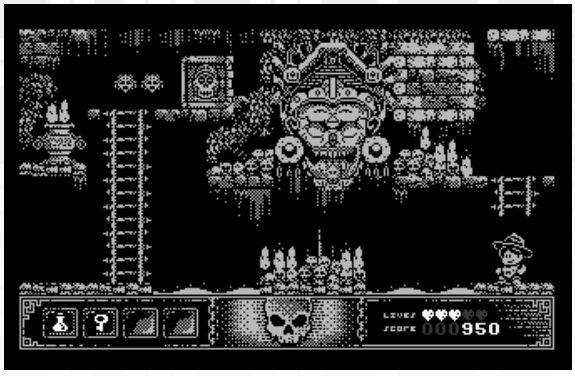
Inside you will find instructions and extras in addition to the media. For author's games and projects where we have a significant involvement, we will include author's dedications with their autographs (you asked for this and we often used to put signatures on disks or envelopes - now we have a better idea for this).

We will publish the games in English. This is about universal packaging and cost containment. We can see from working on the magazine how hard it is to publish two language versions and how much it slows down the whole work. We are therefore proposing boxes and instructions in English, although we will of course also include a Polish localization version in the game itself, as in the case of Die Kaufmannsgilde, whenever possible.

We will keep you updated on our long-term release plans via our social media channels. Today, we can reveal the secret and present the first productions available in our shop. We start with the original game Party Speedway Extd. (more about that on page 58), which we are releasing on a floppy and cartridge. The second production is a two-pack of games from the Windigo Productions studio, containing games from the genre of turn-based economic strategy games (we write more about this project on page 51). It will be released on cassette and floppy.

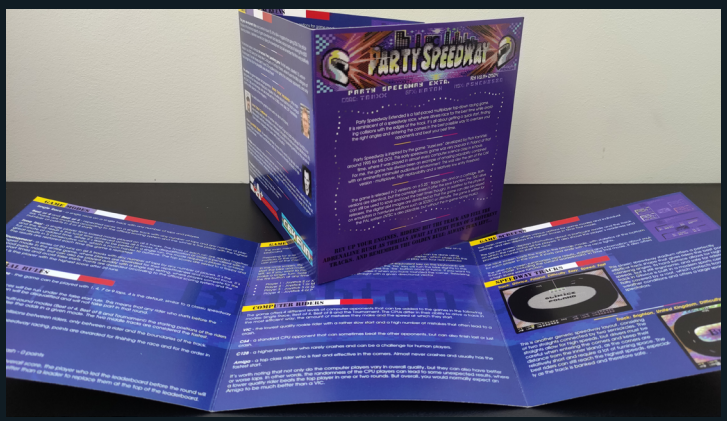
Our next projects for the C64 are a re-release of Lazarus with memories and autographs from its author, Krzysiek Augustyn, and Tony: Born for Adventure, which promises to be a big hit. We are very proud that it was written by our editor, Maciek "Void". The first game we plan to release for the Amiga is an extended version of Ami Hero by Pawel Tukatsch.

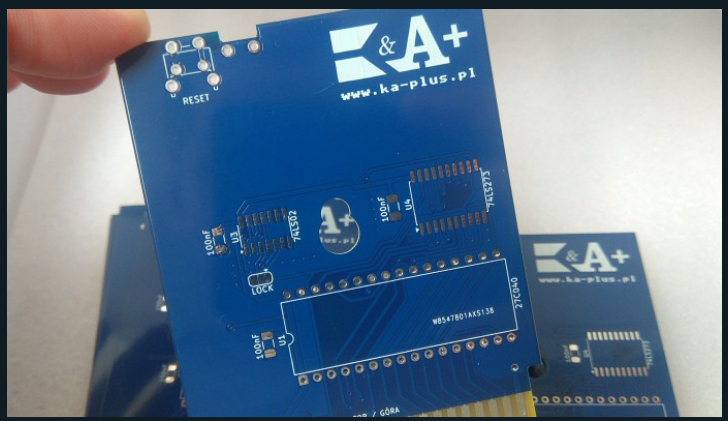
Have a look at our gallery of boxed editions. We also encourage game authors to work with us. ■

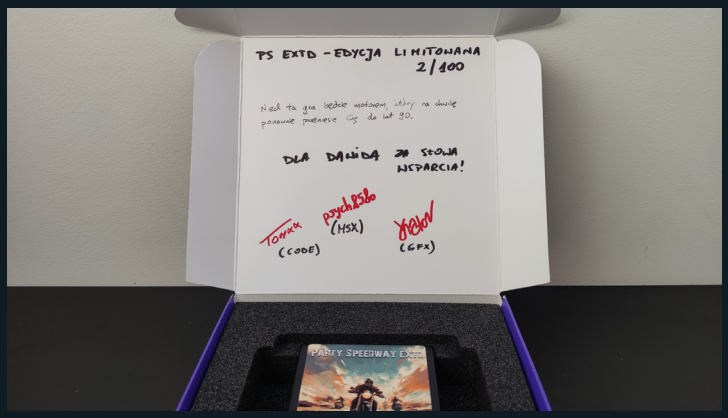


Gallery: K&A+ Collector's Edition

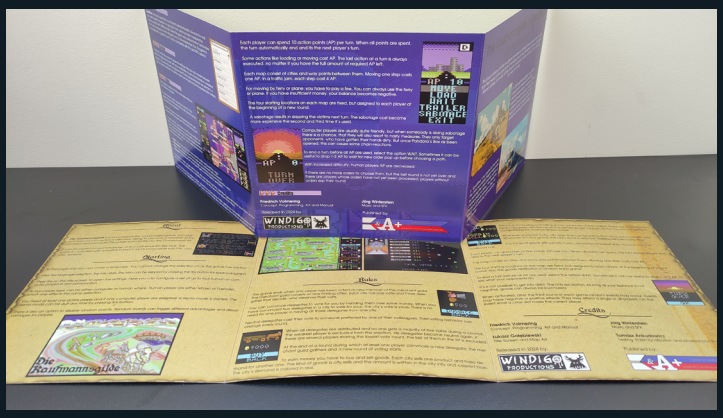
▼ *Party Speedway Extd. The boxed edition contains: universal foam to hold either a 5.25" floppy or a cartridge (a single box may also contain both of these options), a 6-page manual printed on thick coated paper, 2 postcards and personal autographs from all 3 game authors.*

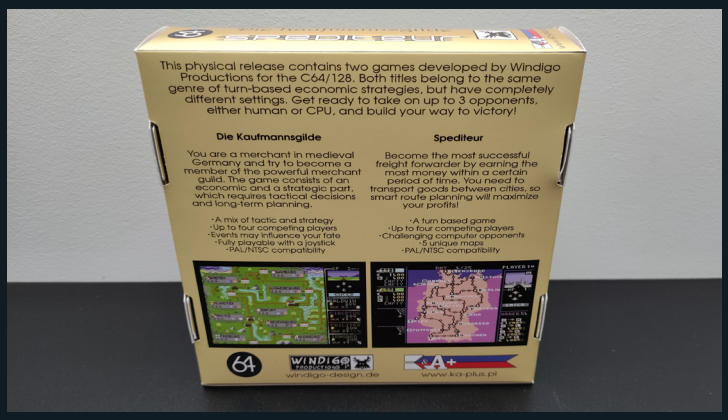






▼ *Windigo Production economic strategy games for the C64: Die Kaufmannsgilde and Spediteur. The boxed edition contains: universal foam to hold either a 5.25" floppy or a cassette (a single box may contain both options), a 6-page manual printed on thick coated paper and 2 postcards.*





▼ *Future C64 releases planned for Q3 2024:*
Re-edition of Lazarus by Inflexion Development, with autographs and personal reminiscences by its author, Krzysztof 'Zephyr' Augustyn.

Tony: Montezuma's Gold - an upcoming adventure platformer by Rafał Dudek and Monochrome Productions (C64 code by K&A+ editor Maciej Małecki). The presented game box is still an artistic concept vision.

COMMODORE

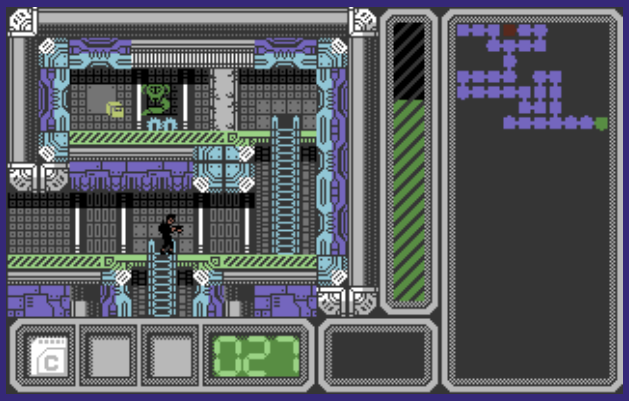
Harharagon

author: Louie Dimovski

REVIEW

What was I thinking about getting involved with Nico? Yeah, she's beautiful and very smart but what chance does a lowly soldier of the realm have with the galactic king's daughter. The king was never going to accept me as being good enough for Nico, which is why he set me up. Damn! As I sit in my lonely prison cell feeling sorry for myself, an apparition appears before me. It's Nico, she has come to help me escape and provides me with a weapon to make my way out of this prison hell.

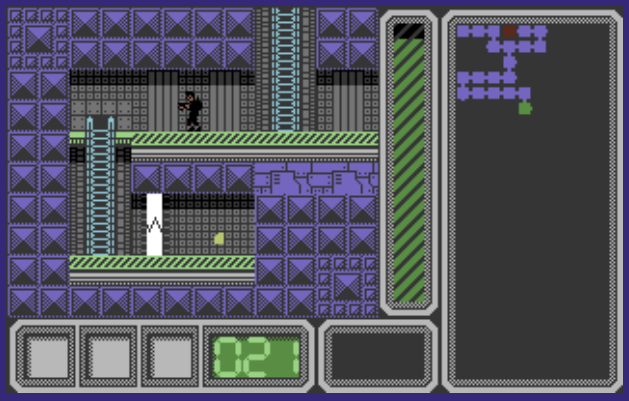
Harharagon is a run'n'gun style of game where your objective is to take control of soldier 64K as he traverses across 230 rooms fighting a large array of enemies as he tries to locate 18 pieces of an escape pod. Many of the enemies you eliminate will drop crystals. Collecting these will be key to your success, as with every 200 crystals you gather, you will receive a very much needed upgrade to your fire power.



Some sections are locked off by gates, which require you to go and search elsewhere to locate the corresponding key card to unlock. Other sections of the game are blocked off by concrete walls and these can only be removed by obtaining the shooting power upgrade when you collect 600 crystals.

There are other useful items to collect such as health upgrades and temporary shields and you are going to need them as you delve deeper down the complex. While the initial enemies are quite easy to combat, those found towards that latter part of the game will not only be much tougher to deal with but they also inflict more damage upon your health bar if you do get hit.

Harharagon contains a great automap feature which is invaluable for showing which part of the prison you have already uncovered and to indicate what items are still left in those rooms for you to pick up and the location of the gates that need unlocking.



The production values within the Harharagon are good. The hi-res graphic style is attractive and your character moves around the screen very well and the 8 way shooting mechanism makes blasting enemies an enjoyable process. The main game contains a single looping trance-like thumping SID tune that plays throughout the game but might become repetitive to some.

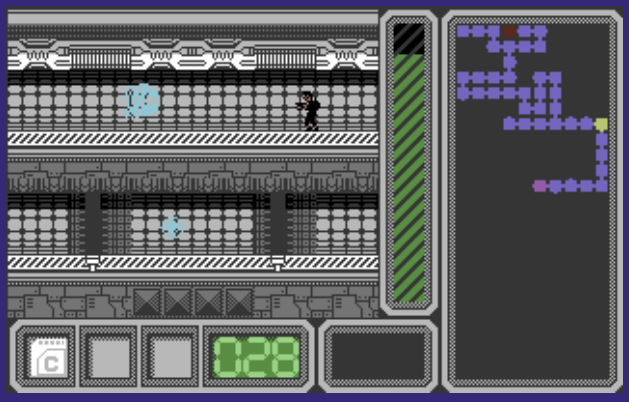
Most C64 gamers are sure to enjoy HarHaragon's easy to pick up and play game style. It does provide a bit of a challenge as the game requires some grinding style game play and strategic use of health pickups to succeed but you are rewarded with further progress with each game you play through. ■

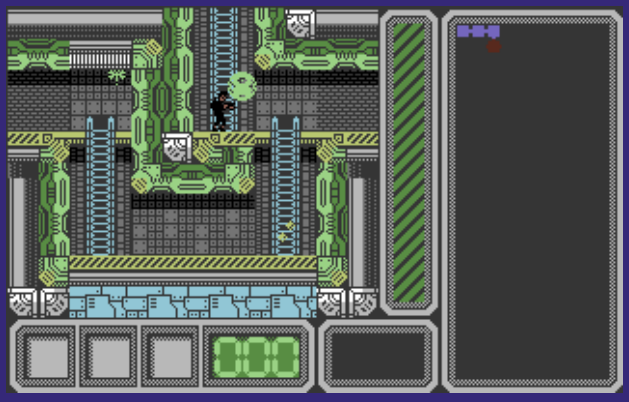
HARHARAGON

Publisher: **Icon 64 / Psytronik**
Release year: **2024**
Platform: **C64**

You got to be a slave to the grind if you want to succeed

8 7 8



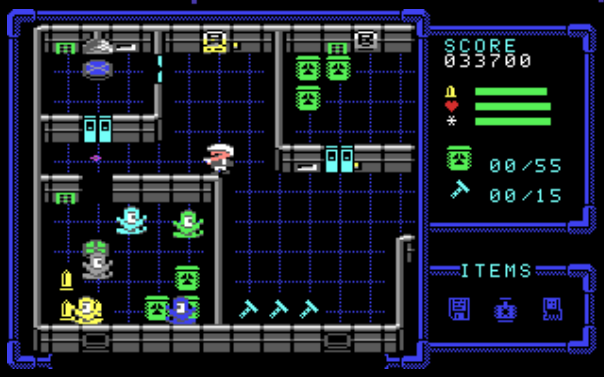


Space Station 23

author: Agnieszka FPWG / translation: Wojtek

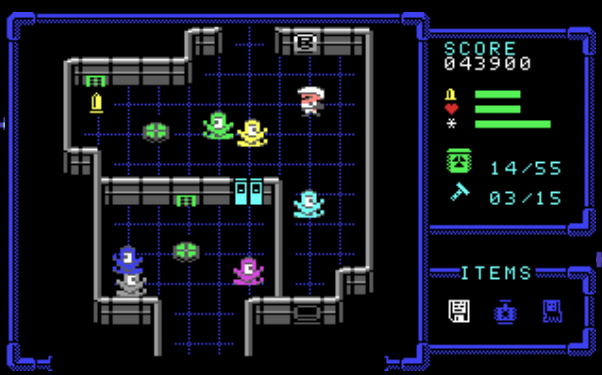
REVIEW

There is no greater nightmare than an alien stalking you in the darkness of space. "Alien" is a perfect example of how a climate of horror and suspense can draw the viewer into a world of madness. Every sound, every shadow makes your heartbeat faster and your fear rise. The alien itself is so terrifying that even the bravest characters lose their minds at the sight of its monstrosity. This is a classic of the genre that should not be underestimated. It is a film that lingers in the memory for years after it has been seen, and undoubtedly deserves to be called one of the best sci-fi horror films of all time.



Alien and Space Station 23 on the C64 share many similarities that make them both terrifying and immersive. Both the film and the game are set in outer space, creating a specific atmosphere and horror vibe. In both cases, players and viewers face an unfamiliar and dangerous environment where lives are threatened at every turn. Additionally, both 'Aliens' and Space Station 23 focus on fighting powerful and incredibly hostile creatures. Both productions feature highly intelligent and deadly creatures that compel the protagonists to make tough choices in their struggle for survival.

SS 23 is an incredible game that provided me with countless hours of entertainment. The player assumes the role of an astronaut who is sent to a space station to uncover its history. Throughout the exploration, the player must solve puzzles, evade traps, and confront dangers. To defeat opponents, the player must manoeuvre their astronaut skilfully, avoid attacks, and shoot accurately. Bonuses and upgrades can be earned during gameplay, which make it easier to combat enemies.



Our main goal is to collect the correct number of blue hammers and green barrels marked with radioactivity and survive the ten stages that take place both outside and inside the station on Venus. While outside, we must monitor the overheating levels, which can result in the protagonist's demise. Luckily, there are machines on each level that can cool down his suit.

The game offers fantastic graphics, surprisingly detailed and colourful. The sounds are an integral part of the gameplay, perfectly complementing the visuals and mechanics. They allow us to immerse ourselves in the game world and feel part of it. The music in Space Station 23 not only adds excitement and dynamism, but also helps us to concentrate and get into the role of a traveller, ready to go beyond horizons and discover the mysteries of the universe.

Space Station 23 is a production that will grab you from the first moment and keep you glued to the screen. Beautiful, colourful graphics, sophisticated sound design, a complex gameplay system and a thrilling story make this space adventure sink deep into your memory and stay in your heart for a long time. I recommend it to all fans of adventure and sci-fi games, as it is a real gem on the Commodore 64 that should not be missed! ■

SPACE STATION 23

Developer: **Vector5 Games**
Release year: **2023**
Platform: **C64**

Save the space station

7 7 7

Yeti Mountain

author: Monka

Isolated disappearances have been reported in the mountains for some time, although these mysterious occurrences have been on the increase recently. Usually they go unnoticed, a person just thinking to himself: "Well, even the best can make mistakes". This time, however, another disappearance is no longer the tragedy of some anonymous person, because it has affected me personally. Chris Piriah, my friend, a ski instructor, a man of phenomenal fitness who has spent half his life in the mountains, has disappeared. It's impossible... I have to investigate this case.

Yeti Mountain is a game that is a mixture of genres given in three acts, in which we have to solve the mystery of our friend's mysterious disappearance. Chris has gone missing in the area of Yeti Mountain. The last place he was seen was around the resort. This is where the whole story takes place.

The first part of the game is a top-down adventure similar to old-school J-RPGs. We try to learn about the disappearance of a friend from the characters we meet. At the same time, we explore the area to find items and clues to help the story progress. In between thinking about where to go and who to talk to, we take part in arcade segments. These involve skiing. Depending on your progress, this will be downhill, slalom or downhill combined with performing tricks.

In the second part, the game becomes an arcade Metroidvania. By jumping on platforms and avoiding enemies, we try to acquire new skills to reach previously inaccessible places and the main suspect.

In the third and final part, we have a dynamic mix of the two previous game modes, with the addition of logical elements.

When I started the game for the first time, I knew that it was somehow a tribute to the game SkiFree by Chris Pirih (so the similarity of the name to a lost friend is not coincidental). It was released in 1991 for the Windows 3.0 platform. I was prepared for action in the style of the ski games I knew from my childhood (e.g. 'Slalom' or 'Olympic Skier').

I really hated those games. I like sports games, even the winter 'Olympics', but I absolutely despise the downhill skiing of the computer variety, where I'm always catching a flag or missing gates, and one little mistake nullifies all the effort and adds to the frustration. I predicted it would be bad, but the interesting intro and adventurous start encouraged me to play.

The authors decided to link some of the texts and images of the characters the player encounters with pop culture icons. On my first attempt, I did nothing for an hour but look for the next interviewees and try to recognise who was who. I didn't think it would be so much fun and that it would have such a good effect on my attitude towards Yeti Mountain.

By the second attempt, once I had carried out the first dialogues, used the items, settled into the ski resort and even learnt the first clues leading to the solution of the puzzle - I was quite absorbed. However, in order to push the plot forward, the arcade sequences need to be completed. So it was time to get on the slopes.

Of course, I didn't read the instructions before I started, and after the first glance at the numbers on the screen I decided I wasn't up to it. However, it turns out that the numbers you have to crank out on the slope aren't that scary and after a few practice runs it's all doable. I was skiing, slaloming, doing some tricks and didn't even realise I was enjoying it.

After skiing and solving an interesting puzzle in the first part, it was time for a completely different stage, where the game presents itself as a pretty good Metroidvania. The action is viewed from the side, running, jumping and solving simple logic puzzles. There is a lot of backtracking, switching levers and acquiring new skills to get to previously inaccessible places. It is at this stage that we make the most important decisions that will affect the end of the game. And there are several, and not all

of them are positive. As a bonus, we also have a side quest, which is optional but rewarding to complete. This part of the game immediately appealed to me.

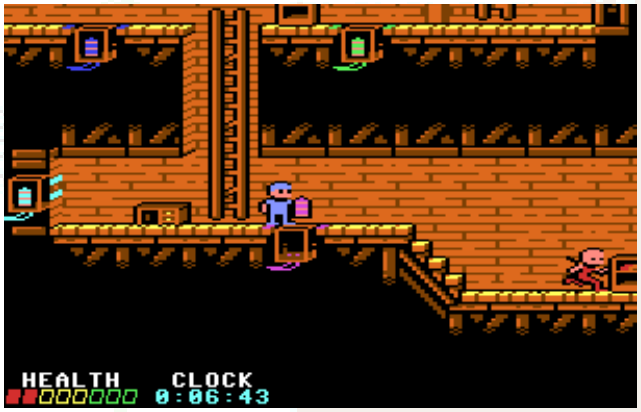
The final stage is a combination of side-view moments and skiing. Here you have to hurry because the action takes on a cinematic pace. Whereas before we might not have paid much attention to time, now we might run out of it. To avoid boredom, new logic tasks and elements on the slopes have been added in this part, which definitely make it easier to complete the game.

The graphics are of a very high standard. From start to finish, we are accompanied by animations in the style of the best NES games. In the first stage and while skiing, everything is clear and easy to read, with very well chosen colours. Conversations with characters are accompanied by nice close-ups. Viewed from the side, there are plenty of details in the background.

The soundtrack is no slouch either. We have great music on the title screen, plenty of music to accompany conversations, ambient sounds in the caves or dynamic and catchy tracks while skiing. There are quite a few tracks.

It took me several tries to complete the game, as I needed time to master the skiing. As it turned out, it wasn't difficult, although it did take some practice, especially towards the end of the game. Skiing wasn't the ordeal I thought it would be. I found it really enjoyable to play.

Praise to the authors for telling an interesting story on the C64 in an unusual style, and for adding a few touches that increase the satisfaction of the game. You also have to appreciate the possibility of multiple endings, which encourage you to complete Yeti Mountain several times and discover all the game's secrets. Highly recommended! ■



YETI MOUNTAIN

Publisher: **Russell Mills / Protovision**
Release year: **2024**
Platform: **C64**

Remember, never eat yellow snow

9 9 8

COMMODORE

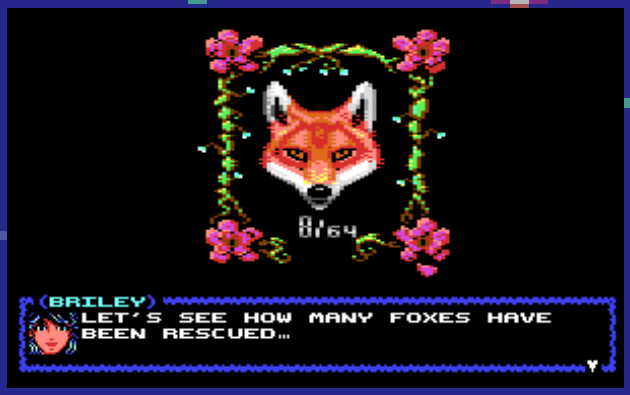
Briley Witch Chronicles 2

author: Louie Dimovski

REVIEW

Briley Witch Chronicles is one of those watershed titles in the Commodore 64 gaming library, easily selling in excess of 1000 copies – a remarkable feat in modern times where your average C64 enthusiast tends to be stuck in the past rather than enjoying what is occurring right in front of them today. Spurred by the success of bringing her novels by the same name across to a digital format, coder Sarah Jane Avory committed herself to providing her loyal fan base with a follow up – will Briley Witch Chronicles 2 reach the same heights as the original title?

The second installment of the Briley Witch Chronicles continues off from the first game and sees you once again take control of Briley as she continues to explore her new life in a world that is still greatly unfamiliar to her. The game kicks things off with a dark and mysterious interaction amongst two cloaked figures conspiring against our teenage witch. Though the intro is somewhat long winded - it is very well presented and builds up sufficient anticipation as to what is in store.



Following the plot of the fifth and sixth Briley Witch novels, Briley Witch Chronicles 2 will instantly feel quite familiar to those who spent a bit of time with the first game. There are a few tweaks here and there and a few new locations to visit but otherwise the game looks and plays pretty much like the first game, including the tedium that is the JRPG combat element that brings the game's enjoyment of exploration and storytelling to a grinding halt.

The story branches on offer with Briley Witch Chronicles 2 are interesting and do a good job of making you care about Briley's plight. These include Briley having to deal with a vindictive Lord Grey who believes that she is responsible for his son's death, the ongoing feud with the Whiteheads, the arrival of a major festival - bringing in outsiders who are not like as they initially seem and a spate of mysterious poisoning that has gripped some of the villagers. I also enjoyed having more involvement with the secondary characters within the game in times when Briley finds herself indisposed within the game.

But Briley Witch Chronicles 2 story narrative is not without its failing. The first Briley game succeeded mostly based on its well balanced and absorbing story telling. But this time around, it doesn't quite feel right for two reasons in particular. The first is that the dialog has lost some of its charming innocence and is slanted even more so towards a female teenage audience and Sarah really needed

So what we really get with Briley Witch Chronicles 2 is more of the same really. Your decision as to whether you want to buy the game should be determined by how much you enjoyed the first game. If you did then Briley Witch Chronicles 2 is a 'no brainer' and a must play title. It is just a shame that it feels more like a DLC rather than a sequel. ■

to translate this dialog to better connect with the middle aged male demographic in mind. The second reason is that the story text dialog is over done in a number of areas and feels bloated and repetitive. I feel that Sarah needed to edit down the amount of text in the game as some of it certainly feels like it's like a whole lot of filler from an 8-bit gaming perspective.

From a production point of view, there is very little to fault the game – it simply oozes quality in this respect. From its in-game graphical representation, the mini facial portraits used to indicate who is talking, its variable music soundtrack, all the way down to the overall game world construct, Briley Witch Chronicles carries all the traits of a professional product that has been put together with a great deal of passion and care.

BRILEY WITCH CHRONICLES 2

Publisher: **Witchsoft**
Release year: **2023**
Platform: **C64**

Sometimes less is more

9 8 7

Grand Prix Circuit

author: Neurocide / translation: Ari

REVIEW

It's 1988 and two legendary Formula One drivers, Ayrton Senna and Alain Prost are racing for Team McLaren. They will win 15 out of the 16 races, only once allowing to be overtaken by Gerhard Berger, driving a Ferrari. McLaren will dominate the season, scoring 199 points in the constructors' standings, leaving their rivals far behind. The world title will be won by Senna, although it will be Prost who will score more points overall. This will be blamed on the regulations, according to which, points from the best 11 out of the 16 races are counted.

Grand Prix Circuit is the best F1 simulator released on the C64. It was created by Accolade, known for many great productions (Test Drive, Fast Break, Power at Sea, Ace of Aces - just to name a few). GPC knocks you out with great graphics, good music and sound, and juicy playability.

The simulator allows you to play in three modes: Practice, Single Race and Championship Circuit. It offers: five difficulty levels; eight original tracks (Brazil, Britain, Monaco, Italy, Germany, Detroit, Canada, Japan); three cars: McLaren - MP4/4, Williams - FW12, Ferrari - 187/88c.

Grand Prix Circuit is a kind of "Test Drive on steroids." In the game we can select a real vehicle from an attractive menu, presenting the car's silhouette, technical data and charts. Once selected, the car starts, and we move to the track. We sit low, close to the ground in the cockpit and see the steering wheel, dashboard and side mirrors. We also have the immortal red dot that tells us the steering wheel's turning radius. And, of course, the gear stick.

As befits a real Formula One race, we first take part in qualifying session (the time we get will determine our starting place). A moment later we can compete against other race participants. At the top left corner we see a map of the track, on which the positions of the player and competitors are marked. On the right hand side the time of the current lap and the entire race is displayed. Revolutionary solutions that are standard today.

The realism of the game is added by the fact that the condition of the car decreases during the race (the level of damage is illustrated by a special bar). Of course, passing the starting line, we can go down to the box to change tires and repair damage.

The tracks are rendered superbly, corners must be taken at the right speed, braking before them and accelerating at the top of them. Even such details as placards indicating the distance between the player and the beginning of the turn have been taken care of. What's more, they are consistently placed on the outside edge . Each track also has distinctive buildings - unfortunately visible only on the horizon.

An added attraction is that our achievements (lap records and race times) are recorded and labelled with the name we chose before the race, or championship.

Of course, the ideal would be the ability to make engine modifications and tire selections. However, the lack of these options doesn't bother us at all. Years ago, the game crushed the competition anyway, shocking with its graphics, presence of the steering wheel and the fact that opponents can be seen in the mirrors. At the time, no one even thought of such additional fireworks.

GPC is, as I mentioned, the best F1 racing on the C64. It's the best racing in general and one of the best simulators of anything ever simulated on this eight-bit marvel. It's also a real treat for anyone who remembers Senna, Prost, Mansell, Patrese, Berger and Piquet. ■

GRAND PRIX CIRCUIT

Publisher: **Accolade**
Release year: **1989**
Platform: **C64, Amiga**

Finally, you can feel like an F1 champion

8 6 8

Press fire to go...

author: Tomxx / translation: Wojtek

EDITORIAL

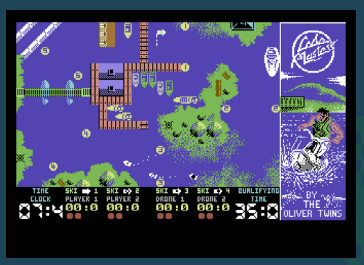
In racing games, vehicle acceleration has historically been achieved in various ways. The most common method was to swing the joystick in the direction of travel or to hold down the fire button (or its keyboard equivalent, such as the spacebar). Depending on the gameplay perspective - from the rear, centre or top of the vehicle - this gave designers different options for adapting the vehicle control. Today, I will remind you of two classic racing games that offer a bird's-eye view perspective. These games use fire to accelerate and tilting the joystick left or right allows you to control the vehicle.

Super Sprint, an Atari classic from the arcades, was ported to the C64 in 1987. Despite the 8-bit machine's limitations, the conversion was moderately successful. While it accurately reflects the original features, the press criticized its playability, specifically the vehicle control and numerous collision detection bugs. Although my memory can be fallible, upon comparing the vehicle control and animation of the two versions today, I must admit that the difference is considerable. While the graphics are functional, they give the impression of being too poor to enchant the player, especially now that Katon is creating pixelated graphic masterpieces for zero-budget racers.

What do I like? The variety of tracks (hills, tunnels, closing gates), the random obstacles and bonuses, as well as the music and the ability to upgrade vehicles. In the two-player mode, this is enough to keep you entertained for a while.

Championship Jet Ski Simulator, on the other hand, is Super Sprint in the water variety. Created by the Oliver Twins (the team behind egghead Dizzy), the game is based on the same mechanics, but with a touch of the inertia characteristic of objects sliding on water. This inertia is annoying at first, but over time it can be used to steer the jet ski smoothly around corners.

The game also offers 1 or 2 player modes and 8 routes. To progress to the next stage, you must complete it within a set time. So, we pass numbered buoys and feast our eyes on colourful, high-resolution graphics of irregular coastlines, marinas, houses, bridges and canals. My fondness for this game is based on the composition of the environment - the bird's eye view, the wonderful combination of the green of the landscape with the blue of the water and the myriad details have always given me great satisfaction. ■



> *The bird's eye view, the wonderful combination of the green of the landscape with the blue of the water and the myriad details have always given me great satisfaction...*

Of the two games, I recommend the latter. The computer level is satisfactory, and you can have an enjoyable race. This is a game in which you progress quickly and become a better version of yourself with each race. ■

Turbo Outrun Plus 4

author: Phowiec / translation: Ari

REVIEW

A long time ago, in a galaxy far, far away... wait, no, that's not this review. Back in the days of 8- and 16-bit entertainment (maybe it was in another galaxy?), people all over the world played on arcade machines or their home computers, just to feel the speed, to chase, to be number one, if only for a moment. Outrun, the Lotus series or Jaguar XJ220 are games I will always remember fondly. I like to go back to them because, despite the passing of the years, they are just as entertaining as they once were.

The aforementioned Outrun lived to see a sequel that was faster, bigger and prettier. Turbo Outrun, released in 1989, did not reinvent the wheel and offered what the previous part did, adding some novelties. There was still a red (of course!) Ferrari there. This time the Testarossa 512 model was replaced by the F40. Adding variety was the addition of a computer opponent, who drove a Porsche 959. And so, racing against time and the computer, listening to music from the radio and having a blonde as a passenger, you sped along the scenic roads of the United States.

At the time, the game was ported to various platforms such as the Amstrad CPC, C64, ZX Spectrum, Sega Mega Drive, Atari ST, Amiga and MS-DOS computers, but there was never a version for the Commodore Plus/4. Until now.

The developers of last year's amazing Commodore Plus/4 ports "The Lemmings" and "Empire Strikes Back" have teamed up again to bring Sega's arcade classics to the old home computer. TCFS was responsible for the programming, Unreal handled the graphics, and Csabo prepared a new soundtrack. The game includes all 16 levels, as well as a special boost feature that allows you to briefly increase your speed.

As befits a 2024 production, the first step is to select the type of control. On the title screen, press return to steer the car with the cursors, spacebar to use the WSAD keys in a new-school way, fire button to operate the game with the joystick. After that, it's all magic: the graphics pleases the eye, the music is catchy, the landscape changes, and you're speeding at breakneck speed across America. Just keep in mind that if our Ferrari goes too slow, Norma, our girlfriend sitting next to it, will get out and leave us. Eh, women....

It will be no surprise if I writ that it is a good game. You can see that the developers have put their hearts into it. It is no coincidence that it was released on January 14, 2024. It's the 40th anniversary of the Commodore Plus/4. Happy birthday, Plus! Thanks to people like the developers of the Turbo Outrun port, I am enthusiastic about the future of this computer. ■

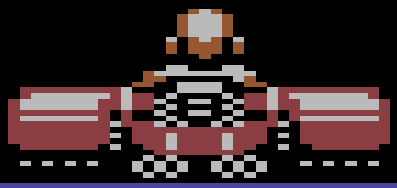
TURBO OUTRUN

Developers: **TCSF,Unreal,Csabo**
Release year: **2024**
Platform: **Plus/4**

The need for speed is strong

8 7 8

Power Drift



author: Łukasz Bobrecki / translation: Ari

REVIEW

In 1989, Commodore computer owners received a conversion of the popular racing game Power Drift. The game, prepared by Sega, gave us the opportunity to race vehicles that looked like a combination of a go-kart and a quad bike. Our vehicle is a machine with a decent engine. The car has only two gears, called low and high, but it can accelerate to certain speeds. The top speed is 224 km/h. It takes a lot of courage to drive such a car!

Twelve people from all over the world have been invited to take part in the race. Before the game starts, we are given the opportunity to choose which racer will be our virtual driver. The choice seems purely cosmetic, but it is worth bearing in mind that we will regularly see the face of our character. Each time we manage to overtake someone, the player in the lead turns to the one who has been defeated and makes a certain gesture. As a toddler, I was convinced that drivers waved to each other in a friendly way. Now I suspect that the cluster of pixels representing a raised hand has more to do with the middle finger.

There are five tracks for a series of races in each of the five available sceneries. To progress to the next track, you need to finish on the podium. A lower position automatically means "Game Over" and the need to start the series all over again. Hardware limitations prevented the creation of 25 completely different tracks, but it cannot be said that the game lacks variety. Races take place in urban settings and in the wilderness, between palm trees and on bridges. On the horizon you may see a ridge of mountain peaks, or a metropolis. To add variety to the gameplay, the designers have made sure that the routes not only wind horizontally, but also have a bend in the vertical plane. In the game, there are regular moments when the track goes uphill and after a while it goes downhill with a sharp turn. If we fail to control the car and bump into an environmental element or another car, we also face a 360-degree turn. Your head could spin!

While driving, the speed at which our car is moving is noticeable. As you accelerate, trees, billboards and buildings disappear as quickly as they appear over the horizon. To add to the experience, catchy, dynamic music blares from the speakers. The amount of sound effects in the game has been kept to a minimum. In terms of graphics quality, there is really nothing to complain about. Considering the capabilities of the computers at the time, there are a few elements that are not found in other racing games of the era. First and foremost are the aforementioned uneven terrain and the eloquent gestures of the drivers. Also worth mentioning is the list of positions held by each driver, which is displayed on the screen at all times and updated in real time.

Sadly, there is no multiplayer mode, and despite all the bells and whistles, racing against a computer opponent can quickly become tiresome. ■

POWER DRIFT

Developer: **Activision**
Release year: **1989**
Platform: **C64, Amiga**

Ladies, gentlemen, pedal to the metal!

7 5 8

C64 racing games review

compilation 88 (W.Czajkowski)

author and translation: Tomasz „RAZOR" Kaniecki

EDITORIAL

I won't hide the fact that the topic of the current issue of K&A Plus, agreed upon at the editorial meeting, suited me very well. Racing games on the commodore were part and parcel of my career as an 8-bit gamer from the very beginning. The cassette number 88 titled "Formula 1 Racing" was the one that showed me what kind of fun this type of games can give.

Looking at the list below, you might be wondering why there are so few of these games - so allow me to indulge you. First up on the list is Test Drive 2, which comes in no fewer than six parts. I will devote a little more space to it in the review.

1. TEST DRIVE 2: THE DUEL (1989)

This game is probably known by everyone. This is the second part, in which we take part in an illegal street race (I have the impression that the PC series Need for Speed borrowed a lot from TD2...). Players will have at their disposal the super cars of the era, from Porsches to Ferraris

and American muscle cars. The cars are controlled from the driver's perspective, and the visuals are very good, with moving speedometer hands, shift animations and other cars in the mirrors.

The tracks vary from mountainous - full of bends, to monotonous - desert. As you drive, pay attention to the road signs - with speed limits and information about bends. The car is not armoured, so it can be damaged if it collides with another vehicle or hits a barrier. An interesting feature is the radar detector - if you exceed the speed limit,

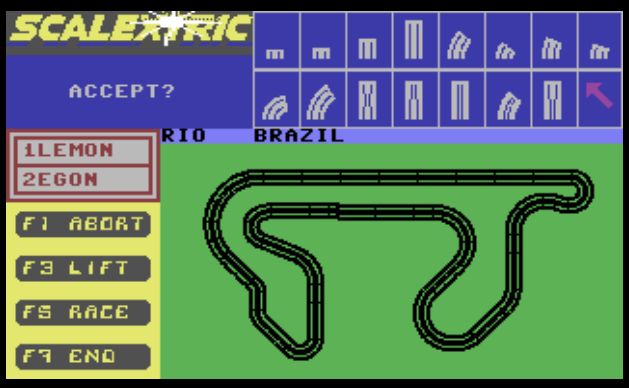
it can start beeping merrily, which means that a police car will appear behind you. They will try to stop you and give you a ticket (just be careful not to drive into their boot when they overtake you, as I did). A fine means the loss of precious minutes that determine who wins the duel. It's also worth mentioning that you have to stop at a petrol station along the way to fill up your tank, and if you avoid it, you'll lose points at the end of the race. I can recommend Test Drive 2 with a clear conscience.

2. CONTINENTAL CIRCUS (1989)

A very cool offering from Taito that gives us the chance to race for the title of Formula 1 champion. The game is a port from the arcade where special 3D glasses were used. No such hardware was available for the C64 version, but the port is well done. We have a full range of tracks scattered around the world. The races are challenging, as are our opponents. We have to pay attention to the condition of our car, bumps will cause the engine to smoke, which will lead to a crash and the exclusion of the car from the race. We have to visit the pits from time to time to improve the condition of the car. The game is very playable, the graphics do their job and so does the soundtrack. It is worth playing.

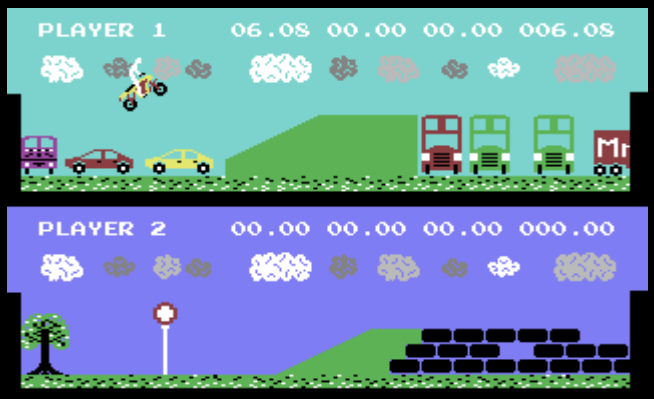
3. SCALEXTRIC (1985)

In the 1980s, plastic racing circuit sets were the dream of many young Formula 1 fans. They were available in every toy shop in the West, but not necessarily in Poland (I still remember playing with such a set with my friend Janek, who got it from his parents' visiting friends). And so the game is derived from the plastic original of the same name. In the Scalextric game, we build tracks from various elements related to the company's existing sets. As a rule, I'm not a fan of games - commercials (the famous years of the industry crisis, remember E.T.?) where the value of the game is subordinated to product placement (yes, we walk around as a bottle of a well-known brand of ketchup and eat tomatoes - sic!). Scalextric, on the other hand, isn't so bad - we can build tracks from scratch, or race on existing ones. The visuals are impeccable, the track-building theme is a cool addition, and we can play one-on-one or alone. Any racer with a flair for construction will like it.



4. KIKSTART (1984)

A game of a legend in a way. I had the opportunity to meet its incarnations on 8 and 16-bit platforms, from consoles to computers. Motorcycle and crazy obstacles, that's the whole recipe for a good time.So we jump our cross over buses, walls, trees and other high obstacles. I don't think I need to convince anyone that this type of game works best with two players - on a split screen. Also recommended.

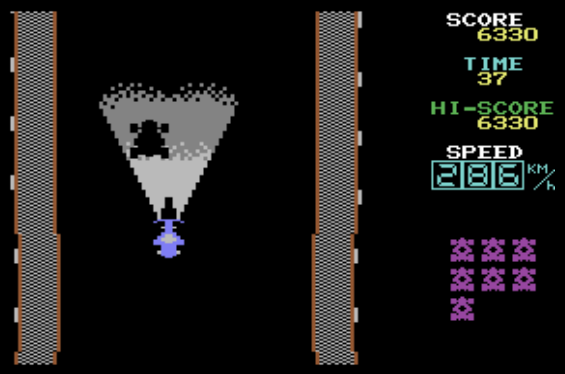


5. 500 CC GRAND PRIX (1987)

Motorcycles on the track this time, but unfortunately not very well done. The game looks hastily put together, the sounds are weak, the animation of the bikes and the track looks like it was made on an Atari 2600 - which is to say, not at all. The controls, some of which use the keyboard, are also overcooked, and that was the nail in the coffin. Play at your own risk.

6. LE MANS (1982)

Who doesn't know the famous Le Mans race? In this game we can see what it would be like. It is worth mentioning that there was a special version of this game where a paddle was used to control the car. Fortunately, I had the joystick version on my cartridge. Visually, it's nothing to write home about, but it's not particularly ugly either. We have races at night, during the day, on wide and narrow tracks, on wet and icy tracks. The game is addictive and challenging. And I don't know about you, but when I look at the screenshots, I see the future of Micro Machines.

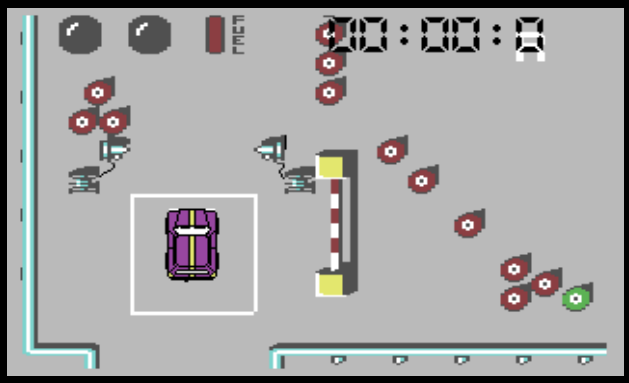


7. SUPER MONACO GP (1991)

Another game originally from Sega ported to the Commodore. Monaco racing hardly needs any introduction. On a visual level, it makes the most of the C64's capabilities. It's a title well worth picking up, offering a driver's perspective, a choice between automatic and manual transmission and, most importantly, the chance to race around the world's most famous circuit.

8. AUTOTEST (1990)

This game doesn't quite fit the bill, but since it's on tape, we'll spare it a few words. In fact, I'd recommend that you start discovering games on tape number 88 with this title, simply because it's a virtual driving course. We learn to park, drive in reverse, turn and brake. The aim is to carry out successive manoeuvres in the manoeuvring area within the allotted time. All to get a virtual driving licence. You have to demonstrate good joystick skills, which can be a bit frustrating at times, but I think it's worth a try just to have some fun.



9. STUNT BIKE SIMULATOR (1988)

Being a stunt motorbike rider is not easy. You can find out for yourself by playing Stunt Bike. It's a single player game in which we perform increasingly complicated and crazy stunt circuits. We have to jump through flaming tyres, climb on the back of a moving truck, avoid oil slicks, etc. You can have a lot of fun swinging the joystick in all directions.

As you can see,there's plenty to choose from on cassett 88. , We'll have the opportunity, if only for a moment, to become anyone from a stuntman to a Formula 1 driver . Most of the described titles give as much fun as they did 30 or 40 years ago.

They take us back to the time filled with thrill, when we waited transfixed for the long awaited "READY" word. To the time where RUN command would transport us into another world, world of winners, pouring champagne, and beautiful hostesses presenting the cup... ■

BreakThru

author: Retrobajtel / translation: Wojtek

I've recently become very fond of arcade games.... I'm constantly discovering and catching up on titles I've missed. I imagine walking into an arcade with a pocket full of quarters. Today, the slot machine is mine.

REVIEW

I can recognise the logos of well-known manufacturers from afar, and I can tell whether a title is going to be good or not. I see a machine with the Data East logo and the BreakThru game. The light dims, and in an instant, I am transported back to my room where I hear my Amiga's floppy drive whirring away. The loading light goes out and what appears in front of my eyes is the same game, or rather its modern conversion from the slot machines. For those who are unfamiliar with this game, let me introduce it to you! You are behind the wheel of a military vehicle and on a mission to recover the super-secret PK430 fighter, which has recently been stolen from us. We charge fearlessly into enemy territory, unleashing a hail of gunfire from the powerful automatic weapon mounted on our vehicle, taking down any and all opposition in our path. Along the way we will encounter enemy vehicles equipped with machine guns, turrets and soldiers who will shoot at us without warning. We will have to jump over various obstacles such as broken bridges, avoid falling rocks and other roadblocks. Every now and then something will fall from the sky, doubling or even tripling the firepower of our cannon. A typical automatic scrolling shooter. What can I say - it's all about reflexes, concentration and smashing "fire" button alternated with jumps.

The game was created using the Scorpion engine, and I must say that it is very close to the arcade original, which as we know, was written in Assembler. I compared the two versions and tried to find some differences and shortcomings. Unnecessarily - because there are practically none, and if there are, it's to the Amiga's advantage. But more of that later. A lot of respect goes to the designers, because it's practically the same game. Of course, nothing can replace the charm of a real arcade machine, but having said that, nothing can replace the quality of Amiga sound chips as well. Although the initial version of the game lacked sound, the current release more than compensates for it. I must admit that the same melody played on the Amiga is somehow more pleasing to my ears. It just sounds better, more professional. The graphics are almost the same. All the details, the animations, the explosions, the bouncing of the wheels on the bumps are taken straight from the arcade version. I even compared the textures, counted the furrows in the rocks, the shapes of the obstacles, their arrangement, and it turns out that they are identical. Well, maybe the graphics are a little smoother on the Amiga, but that's just an improvement. There's no need to split hairs - it's a great conversion. Shoot, dodge, jump, shoot again, speed up, retreat. Ah, the nerves... It's a game for the toughest guys. It's a one-way ticket: either you return in the recovered fighter jet, or you don't come back at all. ■

BREAKTHRU

Developer: **Acidbottle**
Release year: **2024**
Platform: **Amiga ECS/OCS**

Excellent port, pleasant entertainment for the evening

7 8 7

AMIGA

Lotus 1-2-3

author: Jackal / translation: Juan Calderon

EDITORIAL

Which one of us didn't dream, as a child, of driving a sports car, swallowing mile after mile as the engine roars? Shifting gear after gear, leaving the blurry landscape behind while being the envy of every other driver?

Car simulators have always fascinated me, in particular those in which you sit in the cockpit of a Formula 1 racing car or other sports car. As a youngster, I would watch with jealousy as teenagers, sitting at arcade machines, controlled the steering wheel and changed gears, the landscape on the screen rolling past at a breathtaking pace while positions switched back and forth. The first time I could take part in such fun was in the famous Road Race on Atari. However, it wasn't until I got my Amiga that the doors of proper four-wheeled entertainment opened wide for me. To this day I can remember with precision an entire night, admittedly spent on a borrowed Amiga, lost alternating between North and Sound and Lotus II, only ever having a break to enjoy Music Dream II by the famous Swedish group Phenomena. Since then, all three of the Lotus games have occupied a place in my memory from which no other car games, no matter how good, can push them out. Let's take a closer look to these gems, which, despite the passage of time, still manage to both entertain and awaken a strong nostalgia.

LOTUS THE CAR

It's 1976, and the first model of what would become the dream of many car anthusiasts has just left the British factory in Hethel. The S1 version has a 162-horsepower two-liter, four-cylinder R4 engine, capable of accelerating the vehicle to 100 kilometers per hour in less than 7 seconds and reaching a top speed of 221 kilometers per hour. It is no wonder that we will see them featured in notorious Bond flicks in 1977 and 1981, as well as in other famous films. The following years saw the release two further relevant models. 1980 was the year of the S2 - an even faster and more powerful version with a 2.2-liter engine - and in 1996 the world finally saw a 355-horsepower 3.5-liter V8, reaching 100 kph in less than 5 seconds. Interestingly, Lotus was supposed to have a V8 engine from the beginning, but financial considerations led the car maker to discard this possibility, condemning the racing car to four cylinders for two decades.

The vehicle's unique styling ensured it would remain etched on the memory of its owners; in total, only 11,000 of all Lotus versions were produced over the 28 years long journey of the brand. The simplicity, beautiful proportions and peculiar Italian style, as well as the concealed headlights all contributed to its charm, producing a wonderful end result. Lotus' shortcomings in power when compared to its competitors (we're talking about powerhouses such as Ferrari, Porsche and Lamborghini) were successfully made up for with lighter weight, optimal body balance, and - most importantly - a much lower price, which translated into availability and market success, and thus helped build its legend. In 1996, Lotus was taken over by Malaysian Proton, followed by the Chinese Geely in 2017, but that's a topic for another story and, in fact, completely different cars already.

LOTUS ESPRIT TURBO CHALLENGE
GREMLIN GRAPHICS, 1990

Jump forward to 1990, and Lotus is already a recognizable brand in the automotive market. Gremlin Graphics publishes a game by an already well established British studio, Magnetic Fields. Known in the 8 bit world from titles such as Kickstart Off-Road Simulator, it would later increase its fame with the equally excellent Super Cars series. Taking the role of the driver of the turbo version of Lotus, we get to race on 32 different tracks inspired by as many countries of the world, divided into three levels of difficulty: 7 easy, 10 medium and 15 hard. Along the way, we will learn that -contrary to popular belief- there's some excellent highways in Antarctica, while Russian roads seem to be plagued by ubiquitous roadworks. We start as one of 20 drivers, and here is an interesting fact - depending on our score in the previous stage, our starting position also changes. So the worse we drive, the easier the game becomes in the next stage. Conversely, a victory automatically places us in the last starting position, having to take the challenge of overtaking all 19 opponents. The names and surnames of our opponents are based on famous Formula 1 drivers - so we can race against such giants as Nijel Mainsail, Alain Phosphate or Ayrton Sendup and, most importantly, win! We get points for placements, which count towards the final classification and determine the final victory.

Characteristic for this version of the game is the bewildering choice of colours available for our the vehicle - you can have any, as long as it is red. The game has no damage mechanics implemented (such as in the no less famous Jaguar XJ220), and the various hazards in the form of road signs, other vehicles or rocks and trees only slow us down, obviously reducing the chance of victory. On the other hand, we have to watch over the state of our fuel tank, pulling over at appropriately marked pitstops to refill the liquid that gives life to our vehicle.

The game screen is split horizontally and during the two-player game we can observe each other's struggles simultaneously, while the single-player race features an animated image of a workshop on the lower part of the screen. During the game, a car radio allows us to choose to between sound effects or a soundtrack consisting of four well-chosen and upbeat tracks. I must add that the music in all of the Lotus games is an absolute masterpiece for me. To this day, it still holds an honorable high place on my driving playlist and I don't think it will ever drop from it.

All in all, a very good car game and also the most forgiving part of the trilogy - it's quite easy to make up for the effects of your mistakes, provided you don't make too many of them. A remarkable sensation of speed, pretty good vehicle physics and a rewarding difficulty level resulted in quite a success when the game was released.

LOTUS TURBO CHALLENGE II
GREMLIN GRAPHICS, 1991

Almost a year after the release of the first part, the world welcomed its follow-up. This time we have a choice of two vehicle models - in addition to the known Esprit SE turbo version, we also have the Elan convertible version. Already at the very beginning we are greeted by a track by Barry Leitch, which would soon afterwards become a sort of cult hit along with being the one which defines the musical style of the series the best. Both cars' technical data is presented in the form of a neat screen where we can read about the equipment, dimensions of our vehicle and engine parameters. The Esprit encouragingly raises its lights and winks at us, which instantly buys me every time.

The differences between the first and second parts are significant. The lap model has been abandoned in favour of checkpoints, which we must cross within the allotted time in order to keep playing. This time the screen doesn't get split in single player mode, only when two people play on one Amiga and... multiplayer is a blast! The game greatly improves when played in company, which I highly recommend. The difficulty level has been slightly reduced, along with the choice of playable stages. This time, instead of hardly distinguishable tracks in individual countries, we have themed stages. So there is, among others, a race in the forest, in the city, on the highway, in the swamp and my favorite - during a storm. The courses differ markedly from each other, both in terms of the obstacles to overcome and the degree of difficulty and surrounding scenery. In my oppinion this is a significant step forward, as the stages have gained a depth and atmosphere which was lacking in the first game. Iit also means some tracks may be more difficult for one person and less for another - just like in real life, some people may feel more acquainted with driving on a highway than amidst the fog. We can access any course independently through the use of codes, and there's also a hidden mini-game accessible by entering the code DUX. The inquisitive may also find a code which disables the passage of time in the game, but then the gameplay loses a little sense....

During the game we are not accompanied by any music, as it appears only in the menu, at the end of the game (these are mega-catchy tracks, one of my favorites), and as introductory theme songs when loading routes - short, but also well-chosen and pleasing to the ear. Along the route, all we hear are sound effects (engine, hitting obstacles, screeching brakes, thunderstorms, etc.), including some motivational high-quality digitized voices confirming that we've made it through a checkpoint, collected a turbo, and the famous "congratulations!" when we cross the finish line.

As in the previous part, there is still no damage model, and as a result the game is not stressful. Traversing kilometers has been spiced up with various easter eggs and a really cool ambience (I recommend driving UNDER a truck driving across the road on the highway stage). The snow falling, traversing through roads in a thick as milk fog, speeding towards the city at night - all these deep memories built my strong nostalgia for this part.

> *This journey through the Lotus series is a true hurricane of nostalgia for me. A memory of a time with no internet and no manuals, when me and my friends would try to figure out, with flushed faces, what a game was about based on its name alone. Lotus was very straightforward to introduce: it was exactly what a kid raised on model cars, Bburago and Matchbox catalogs expected it to be. Not only that, but as I found out once again in 2024, none of it has aged a bit.*

Lotus 2 is a very high in my ranking of the best Amiga car games. Starting with the soundtrack, through the atmosphere, ending with the difficulty level - everything gives the impression of a well thought-out combination and always evokes in me the fascination from my youthful years, when Lotus was one of the most solid choices for multiplayer gaming, and those nights spent with the game will remain for a long time as a landmark of good fun.

LOTUS III – THE ULTIMATE CHALLENGE
GREMLIN GRAPHICS, 1992

Just a year had gone by, and the culmination of the trilogy arrived - in pretty good style, if you consider the time it took to produce it. If you were expecting some variety, I'll disappoint you - the last part of Lotus is not a groundbreaking novelty, as it is basically a combination of the previous two.

The very intro to the game gives a sense of closure: the escalating sound theme that builds tension and the Lotus coming into the center of the screen is in a class of its own. Astonishingly, decades ago, skipping the introduction was not a thing - the beginning even built the necessary atmosphere.

Returning to the game, all its elements give the impression of being better, more polished and, I remark, definitive. We have both a time trial and a championship, in which the best of 20 drivers wins. There are three vehicles to choose from, as a new car has been introduced to the game - the M200 prototype, a version of the Elan M100

speedster, which was presented as a concept at the 1991 Frankfurt Fair and the 1992 Geneva Fair. Unfortunately, this version never made it to mass production, and the only existing copy is in the possession of a private collector. A story worth exploring in itself.

Several new stages have been added in this third installment of the game, so we have the opportunity to race in the mountains, under strong winds, dribbling around road works on the highway, on gravel and - as icing on the cake - in a distant future vaguely inspired in the Tron movie.

Once again, we can listen to music during our rallies. We have six new tracks, dare I say much better than those from the first one. It's an absolute must have for late-night car rides on long journeys, plus each of these tracks has been remastered and mixed dozens of times over the years since the game's release by top Amiga musicians, which is a testament to both their popularity and their quality. In addition, before each stage of gameplay we get to listen to a catchy introductory theme, just like in the second Lotus.

The third installment of Lotus received criticism for recycling elements from previous versions; it was basically a content update, lacking of new and creative elements. Surprisingly, what was its biggest flaw according to these critics turned out to be exactly what fans expected - more of the same. What's more, even more strongly than in the case of two, the game showed its true claw in multiplayer. Thrilling two-on-two duels were the norm in the distant 1990s, and all this (plus a simplified RECS track editor) resulted in high playability. I keep coming back to this version of Lotus more than any other.

This journey through the Lotus series is a true hurricane of nostalgia for me. A memory of a time with no internet and no manuals, when me and my friends would try to figure out, with flushed faces, what a game was about based on its name alone. Lotus was very straightforward to introduce: it was exactly what a kid raised on model cars, Bburago and Matchbox catalogs expected it to be. Not only that, but as I found out once again in 2024, none of it has aged a bit. The graphics, the music and the pleasure of playing each part is timeless. So... on your marks, get set, go! See you on the track. ■

Console zone

RGCD Racers V1

author: Don Rafito / translation: Ari

REVIEW

Amiga console compilations are a really cool solution. Whether we run them on physical hardware or on an emulator, there is no need to install the entire AmigaOS. We just insert the disc or mount its image and we can enjoy pure retro entertainment.

Another advantage is that compilations for the Amiga CD32 are often divided thematically, and every fan can compile his favourite categories of games. The creators of these compilations are usually fans themselves, but sometimes they are signed by retro-gaming publishers. This is the case with our hero today.

RGCD Racers Volume One is a compilation of racing games created by Heavy Stylus using, among other things, slaves for JOTD's WHDLoad package. The CD contains four carefully selected racers that are well known to Amiga amateurs. They are:

All Terrain Racing (**Team 17, 1995**)

Micro Machines (**Codemasters, 1993**)

RoadKill (**Acid Software, 1995**)

Turbo Trax (**Arcane Entertainment, 1995**).

All the titles feature a bird's eye view gameplay. The compilation's creator also focused on ensuring that the selected titles would run well on a standard Amiga CD32. A simple and uncluttered menu appears on the screen, showing the game's logo and a short description of which button has been assigned to start a particular game. The graphics of the main menu are clear, but it lacks a dynamic-sounding, car-compatible module.

Games start without a hitch, but there is a slight problem if you have a turbo card with a faster processor. After holding down the fire button, the title screen of the selected game would appear for a moment and then everything would return to the main compilation menu. Finally, the game would only start after a few attempts.

Although this compilation is a few years old, the games it contains are even older. But they are still the best Amiga classics among the racing games. So if you have an Amiga CD32 in your collection or emulate one, this kind of compilation is a very good solution. No system installation and no complicated configuration - just boot up the console and voila! Pick up the gamepads and off you go. ■

RGCD RACERS V1

Developer: **Heavy Stylus / RGCD**
Release year: **2011**
Platform: **Amiga CD32**
Rating: **9**

AMIGA

DEATH RALLY

author: Retrobajtel / translation: Ari

REVIEW

"Gimme fuel, gimme fire, give me that which I desire!! Yeaaah!" The words shouted by James Hetfield over sulfurous riff of the song Fuel, perfectly capture the atmosphere of the rolling on the monitor screen, spectacular race. Only that here it is even more hardcore, here the game is for the highest stakes.

Because it's a race to the death. As the slogans proclaim - life is short, hot and fast here. Be careful on the road, listen to the sound of the engine and working pistons. Let your lungs fill with the smell of exhaust fumes, the stench of burnt oil and gasoline injected into the cylinders... Let others feel the bullets from your cannon pierce their engines and slice their bodywork, leaving burning vehicles on the road like burnt-out cans of canned goods. Another rally, cash in hand and slight modification of the vehicle.

How about a little sabotage before the next race? That guy with the weird haircut is too confident. It will be nice to watch the brake fluid from his slashed hoses pouring onto the road, and how moments later the car hits the concrete pillar of the bridge, hitting a few eager onlookers along the way who chose the wrong place to watch. There is no mercy, no room for a moment's hesitation, the only thing that matters is winning. Such are the emotions I feel after spending a few moments on the deadly and twisted tracks that may be the last one for careless driver.

Death Rally is a port of the 1996 PC game, developed by Remedy Entertainment, those from Max Payne and Alan Wake. It was developed by Bszili, who is responsible for such Amiga ports as Wolfenstein 3D, Blood or Shadow Warrior, for example. Well, so it is already known that we will take no prisoners...

Interestingly, the Amiga version of the game was created using reverse engineering, which means, more or less, that it uses the original PC version files.

Unfortunately on a stock Amiga 1200 we won't be able play. The minimum requirements are a 68040 processor, or preferably 68060 and 8 MB of FAST memory, the presence of AGA chipset on the board is also an obvious necessity.

If your equipment meets these requirements, you will find it hard to tear yourself away from the game.

The first thing that came to my mind was the 2008 movie Death Race with Jason Statham, but unlike the movie, we are not framed into anything, we are not sitting in a jail.... We are simply running races at our own request. Because the money is tempting, because regular driving around the city will not provide us with such impressions.

At the very beginning we have a small amount of cash and a vehicle resembling a "VW Beetle". There are a dozen tracks, divided into three difficulty levels. The higher the level, the more money we can earn, but we can't do much with a weak car, so we start with less profitable races to buy a better car over time. So we drive, shoot, put mines behind us and, of course, try to win, either by reaching the finish line first or by eliminating the other racers. Along the way we collect cash and ammunition scattered on the road, which quickly runs out. We also come across other "finds" that, for example, "on the fly" repair our vehicle, or we come across hallucinogenic mushrooms, after taking which the track, or rather our vision, begins to wave. Before and after each race, we can visit the garage, make the necessary repairs, carry out minor engine modifications and arm our car with more deadly toys. Of course, you have to pay for everything, if we are short of money, we can sometimes take out a loan, it may pay off because other players are not idle. We can also order sabotage, such as the aforementioned cutting of brake lines, or perform some order during the race, such as eliminating a particular competitor, for which we get extra cash. The goal of the game is to defeat the main boss in a one-on-one race, but I'm still a long way from that.

The game really does look very nice, and after a long time of playing and the first relatively good race (not finishing

last), it is a lot of fun. The view from above, the dynamic and slightly changing perspective camera and the environmental elements such as trees, lanterns or bridges create a three-dimensional effect and do a great job of providing cool visual fireworks. The music playing in the background, which was not present in the original version, makes you want to press the accelerator even harder. As for the gameplay itself, I was initially a little frustrated by the controls, which took some getting used to. Too often I ended up hitting a wall or driving into a crowd of onlookers. But after winning a few races, collecting a small sum of money and installing a few toys in the car, a smile appeared on my face... And I felt the desire to compete. I also moved up a bit in the rankings, so I wasn't such a newbie anymore, and they finally started to be afraid of me... Just this "VW Beetle"... But I'll buy something better soon.

So show me the new one, the one who claims he can race... I guess he doesn't know what he's signing up for... Here we go! Pedal to the metal, turn quickly and fire the gun. I think the guy was stunned. I pull up to him, roll down the window, see him shaking, tears streaming down his cheeks, and say, "Son... go home. This place is no place for you. He doesn't say a word..." I drive off, kicking up clouds of dust, throwing mine right under the wheels of the guy sitting on my ass... .... A loud crack of falling metal, a short groan and silence... Total silence. Just the smell of petrol, burning rubber and something else, I'm afraid to think what. Yes, ladies and gentlemen, welcome to the death race. ■

DEATH RALLY

Developer: **Szilard Biro**
Release year: **2023**
Platform: **AMIGA, 68040/68060**

Gimme fuel, gimme fire, give me that which I desire!! Yeaaah!

9 9 8

3D Rat Race

author: Don Rafito / translation: Juan Calderon

EDITORIAL

Cars, racing, racers... games. Ever since calculating machines began to delight their users with electronic entertainment, developers have been bringing pieces of the real world into the digital realm. Sitting in front of a screen, we became athletes, commandos, spies, knights, travelers, all manner of creatures and heroes of virtual worlds. Including race drivers.

As technology developed, the processing power of computers increased. The development of games sped up proportionally: a larger colour palette, higher resolution, better sound and increasingly realistic mechanics and physics. The latter two in particular affected simulators - and racing games formed a large part of these. Naturally, car games that were not typical simulation titles also appeared on the market, but any title that sought to recreate real life conditions in any way had to stand out by way of its execution and appearance.

And so, getting down to the subject of the article, the developers primarily opted for a change of perspective. If a given production aims to convey the feeling of driving a racing car, the player must feel as if he or she is sitting behind the wheel. The on-screen action must therefore be seen from a first-person perspective. Other important elements, of course, and not only for racing simulators, were the graphics and animation. They had to resemble reality more and more. Three-dimensionality was the answer, sometimes even at the expense of colours or sound design. Why is this? Because it turned out that a game engine built in this fashion required more resources from the computer than a title based on two-dimensional sprites or bitmaps. Of course, over time, different techniques began to be mixed, combining three-dimensional environmental objects such as the track, bridges, pit stop or grandstands with two-dimensional elements representing signs, trees or other drivers. This allowed for a better colour palette and sound design.

One of the first games of this kind was Turbo Esprit (1986). The game was released on the ZX Spectrum and Commodore C64, among others. It features a first-person perspective, three-dimensional buildings around the course, a road and, of course, opponents. We move constantly around the city in straight sections, occasionally turning into perpendicular streets at intersections, and our task includes overtaking and evading other road users by constantly changing lanes. This game makes a good benchmark for how performance varied depending on the platform on which it was released. On the ZX Spectrum, we have the basic elements of a three-dimensional car game, but the limitations of this home computer did not allow for the use of more colours and sounds, which the C64 version enjoyed. There, the audible sound of the engine and the colourful vehicles of the opponents are the most striking features.

The same goes for another simulation racer. The infamous Test Drive (1987) and Test Drive II (1989), in addition to a sense of three-dimensionality, introduced further innovations, such as traffic regulations and the possible consequences of failing to observe them in the form of encounters with the police. Of course, most of the graphics in the game are bitmaps, but here too you can see the difference between the C64 release and the Amiga. The latter version, thanks to its noticeably greater computing power, features a wider colour palette, better sound and smoothness. And, as is common in this business, 16-bit machines slowly began to displace eight-biters, resulting

in better and better games, simulators that were ever closer to realism, and in particular the racing simulators discussed here. The turn of the decade showed that the Amiga was perfectly suited for this.

This was also evident even when confronted with the fledgling PC game market. For example, Ferrari Formula One: Grand Prix Racing Simulation (1988), which - as the title suggests - is a formula one racing simulator, looked incomparably better on Amiga than the MS-DOS version. It was smoother and better sound-wise, all thanks to the specialised chips. Unfortunately, it was these chips that later sparked the race giving title to this article, which we all know how it ended. But about that later; and in the meantime let's go back to the glory days of our Friend.

Vectors, or the building of three-dimensional models of objects, which were then covered with colours and, later, textures. One of the first titles to use this type of graphics engine was Stunt Car Racer (1989). There you could feel the advantage of the Amiga edition over the PC one - it had better sound, graphics and frame rates. It is worth noting that the game was also released on the C64, and the PC version was very close to it. The Amiga remained on top.

As years went by, so did further games featuring an increasing number of vectors. The Amiga Powerdrome (1989) again showed its class over the PC one, even though the latter version was a tad smoother. However, apart from a few samples it was completely silent. The racetrack, on which the futuristic vehicles ran, was made entirely of 3D wireframe and covered with single-colour textures. Shading was also used: the textures changed when our vehicle flew through a tunnel. The only downside to this racer was the difficult controls. It wasn't as obvious as in a car or a kart, and if you oversteered you could hit a bump and fly off the track. It is clear, then, that the development of three-dimensional graphics in simulators, and cars in particular, went hand in hand with the development of physics and game mechanics.

The next title, Hard Drivin' (1990), showed that a third competitor was joining this race: arcade machines, the ancestors of consoles. But let's talk first about the tracks. PCs are gradually starting to catch up and, in terms of graphics, the titles are more or less comparable. Apart from the soundtrack, of course, as the MS-DOS versions are still lagging behind in this respect. Furthermore, there are signs of further development of vector graphics and 3D engines -not just the track, but also the entire scenery is now built out of a textured model: bridges, loops, ramps and also other cars. There is also a new mechanic,

namely the collision - whether with another vehicle or an element of the environment, sometimes ending the race with a broken windscreen. We can watch it all in replay with a bird's-eye view and enjoy all the 3D scenery, as well as spotting the mistakes in the course of our duels in the tack. And the best in this head-to-head battle was... the arcade version. The 3D models, the textures, the frame rate, the sound, were all very polished and setting new standards for racing game simulators. The second part of this game, Hard Drivin' II: Drive Harder, released a year later, is mainly a repetition of the same idea, but with new tracks. Again, the PC was slightly better in terms of fluidity, obviously at the expense of the soundtrack, while the Amiga game could get slightly choppy when driving around large objects such as bridges or tunnels. This lag is quite subtle, but nevertheless noticeable to the trained eye. Could it be that competition for Amiga's specialised chips in the form of PC VGA graphics cards is emerging? We'll get to that.

MicroProse, the global powerhouse in simulation games, and its production Formula One Grand Prix (1991) showed that the Amiga was beginning to catch up with the PC. The title in both versions was of course made of vector polygons, including the opponent's cars; the sound was more or less comparable, but the fluidity of the animation, the colour palette and the execution of the textures spoke in favour of the MS-DOS version. The race for the title was underway, and the growth of the PC, the increasingly sympathetic eye of developers and publishers in their direction, and Commodore's impending troubles meant that the Amiga was beginning to lose pole position.

1992 saw some good titles in the category, but their releases on both platforms began to diverge to the detriment of the Amiga. Vroom was a return to bitmap graphics combined with a few vector objects, giving the Amiga game a smoother experience. The sound department was also kinder to the Amiga. But the colour scheme was better on the PC release. A larger colour palette and the appearance of gradients made these versions more eye-catching, which was not necessarily synonymous with playability. The next title was 4D Sports Driving Master Tracks - a game in full vector graphics. The colour palette was more or less similar, the audio comparable, but the PC version stood out thanks to its fluid frame rate. Unfortunately, the Amiga's specialised chips, when compared to the VGA graphics cards, were starting to struggle to move the three-dimensional polygons around, and more and more often it resembled a slide show rather than a smooth ride on the track. This was starkly evident in the next entry, entitled Race Drivin'. Even the ECS chipset, which appeared in the Amiga 600 released the same year, did not help. The good old Motorola 68000 was simply showing its age. And in order to play catch up, either a faster and more powerful processor or a better graphics chip was needed, or preferably both. Which, in the example of the title in question, was evident on AGA amigas, where Race Drivin' was clearly shining brighter. Regardless, that was nothing but a temporary patch on the problem. On the horizon, in addition to ever more powerful and popular PCs, another nail in the coffin appeared in the category of 3D graphics - and not just regarding cars. We are, of course, talking about 3D0 consoles. Hover Sprint, as the last released production from

the same year, appeared as if to wipe away the tears. The Amiga edition, although obviously also making use of 3D vector graphics, those were mostly flat, with decent colours, good sound and much smoother performance in comparison with the version released on... the Atari ST.

The next two titles to arrive were, technically speaking, indie productions. The first, Racing Maniacs (1993), was distributed as Shareware. It featured sparse vector graphics, a decent soundtrack and, for an OCS title, reasonably smooth performance. The second, F1 Racer (1994), on the other hand, was somewhat of a copy of Vroom, that is, a combination of bitmaps with a small number of vector objects, and was therefore quite fluid and enjoyable to play, including above average color and sound. Unfortunately, these games did not move the Amiga past the 3D leader that the PC was becoming. In addition, Commodore's troubles and subsequent demise meant that users, and above all developers and publishers, were moving away from the platform. Nevertheless, the race was on and persistent fans and developers who had not yet said their last word stood behind our Friend. The result was productions perhaps not quite aiming to compete with MS-DOS and later also Windows games, but wanting to show: "Look, we have our 3D games too!". And despite the fact that the Amiga was becoming a niche platform - and the arrival of Sony's PlayStation console further sealed this fate - the production of accelerators, graphics cards and all sorts of interesting expansions or even the very existence of AGA based models meant that our computer was still on course. Troubles aside, the Amiga could still have done pretty good, if it wasn't for the delays and the trend to count one's chickens before they hatch. And this goes not only the successive owners of the Amiga, but also for the developers, who were churning out productions that did not make use of the new technological solutions available in an attempt to reach into the pockets of a dwindling number of fans; and those who did try such titles for lack of a better option were not too pleased. Fortunately, not all developers did this, with the result that the Amiga was still in the game despite its weaker position on the circuit. Although, as mentioned above, it was starting to look like a separate race altogether.

Xtreme Racing (1995), despite being done in pseudo-3D and lacking vector graphics, won over many car enthusiasts thanks to its use of the AGA chipset and skillful execution. Although the game world was flat, the textures were well defined with colour and proper shading. This gave it a three-dimensional feel. The enjoyable and vibrant soundtrack and the increased frame rate available for those owning more powerful processors and the multiplayer option were other highlights of this release. All of this made the game simply good and kept you glued to the screen for a long time, allowing you to forget about the racers with complex 3D engines and ever more realistic textures that were reigning supreme on PCs and PCs.

In the fully 3D department saw two more titles were released that year. The first, Leading Lap, featured full 3D vector graphics, covered in monochrome dithering textures - which unfortunately detracted from the aesthetic feel, but benefited a smooth gameplay - a decent horizon background and gradient skies. The track sounds were average, but the musical interludes and digitised coach voice were quite good. The title was received with a hype reminiscing the early 1990s and while it could hardly match the main rival platform, in keeping with the "We've got our 3D racing" principle so far, it was able to repay the favour with decent playability, not least because it offered a multiplayer mode via cable or modem.

The second entry also released only for the Amiga, Virtual Karting, was perhaps best known for being simply mediocre. Apparently 3D, but flat, with poor, blurry textures and poorly defined background or track elements. What's more, even though the game was made for AGA chips and required them, on vanilla Amigas, in order to play smoothly, one had to additionally apply a filter, which made the already poor graphics even worse. On the sound side, the engine whine was mostly average and the tyres squealed a bit, but it was saved by very enjoyable and fast-paced musical interludes. Three years later, a sequel was released in the form of Virtual Karting II. Unfortunately, most of the ideas from the first one were repeated and barely any new tracks were added, resulting in an underwhelming reception of the game, which did not have a positive effect on image of the Amiga as a computer on which to race in 3D.

Although we have already jumped to 1998, let's disrupt the chronology and go back a year. Wheels on Fire was the first 3D car game for the Amiga to use the vortex method. As a pioneer in this field of electronic entertainment on the Friend, it unfortunately did not offer mind-blowing quality. The textures were a bit pixelated and not very clear. Some backgrounds were also missing here. The sound side was done quite correctly, with good quality samples and background music. Despite a cutting edge graphics technology full of shortcomings, which required AGA chipset and recommended at least a 030/50 processor, players appreciated the novelty and some playability. Although, on the other hand, should these hardware requirements be regarded as insane or rather the norm for 1997? The Amiga wasn't retro back then and was still supposed to be comparable to the PC or PS One. And although complaints were made about its playability, poor mechanics and lack of difficulty, a certain game brought our Friend somewhat closer to even the Playstation in terms of 3D graphics.

I am, of course, referring to the game Flyin' High. In addition to the required AGA chipset, the title also supported Amiga graphics cards. Indeed, the quality of the engine and textures was very good. The sound side was quite decent, although some samples or background music seemed a bit tedious. In any case, the title gained a fan following, resulting in the subsequent release of extra content with additional tracks.

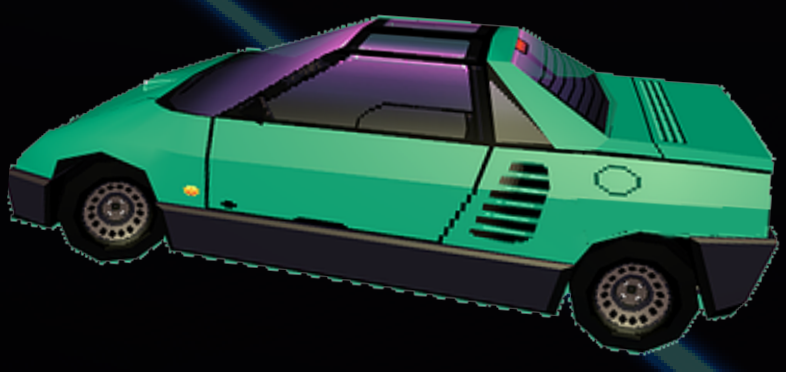
Turbo Racer 3D (1998) also decided to follow the same path in execution technique as its predecessor. Unfortunately, it didn't quite succeed. The textures were more pixelated and less recognizable. The sound side was

rather sparse, as it basically consisted of only two background music tracks, albeit quite good and lively. What stings the most about this title is the mechanics. On one hand it reminds a lot of the Lotus game, and on the other, the feeling of speed is just not there. Although the counter may show 200 km/h, the entire animation and movement of the track feels like driving through a busy exit road on the holiday season. On the plus side, the game can be played in two modes: Lores for the standard Amiga and Hires for those blessed with an accelerator. The game also requires AGA chips and does not support graphics cards.

An example of how Amiga racing games had the potential to compete with, say, early Sony Playstation games, and thus narrow the gap on our technological race, was demonstrated by Virtual GP (1999). A good three-dimensional engine, realistic textures, a decent soundtrack, support for AGA chips and graphics cards and good mechanics were the main strengths of a game which should have set a new standard for driving games. It should have, but it didn't. The modest number of users and publishers - basically hobbyists at this point - were unable to feed the Amiga gaming market. And the still crumbling community of gamers, who could also already be called hobbyists, did not guarantee a large turnover in business. Besides, the Amiga was moving towards PPC processors, and most of the users who still stood by our platform had placed their hopes of staying in the race on this technology. A showcase for this hope was the conversion of Wipeout 2097, released in the same year. The game had previously been released on Playstation (1996) and PC (1997). Futuristic racing with quality graphics and sound and fast paced gameplay showed that the Amiga still had the possibility of being in this duel thanks to the new architecture. Unfortunately, this opportunity was not seized. The legal turmoil in which the successive owners of the rights to our computer became involved caused everyone to divide. Both developers and users. Wars were fought in the courts and on discussion forums. This also forced the division of the development of the Amiga itself, basically already a Next Generation platform, and split it into three streams: MorphOS, OS4 and AROS. Unfortunately, the number of productions for these platforms, despite the new possibilities, was not impressive when it came to 3D car games. There have been, and still are, mainly titles that are ported from other popular platforms, mainly on an open-source basis, and some of the better ones worth recommending are MultiRacer, SuperTuxKart, Speed-Dreams or TORCS.

Well, what about classic line users? Before Amigas were sold off, hidden in the attic or, worse, thrown away, gamers could once again taste the 3D atmosphere that the last mentioned productions gave them. Joyride (2000), although visually a little flatter than VirtualGP or Flyin' High, also provided a decent 3D racing experience. Well thought out mechanics, decent textures and colours, and an upbeat soundtrack made you want to get behind the virtual wheel more than once. And then there was the feeling that something was still being done, that the Amiga wasn't out of the race, albeit on a track of its own.

Now, when retro fashion has taken over, the Amiga has gained new fans and some users have returned, there is a chance for this race to continue, although - as you can see - it has split into many separate tracks. Besides, today's Amiga, especially the classic one, has gained possibilities that were not there at the end of the 1990s, thanks to all the products that have appeared in recent years. Warp, Vampire, PiStorm, WinUae, A500Mini - do we need to mention more? No, these should be taken advantage of as a priority. It is known that this is a niche market after all, but the race is on. And productions such as Vroom, Xtreme Racing, VirtualGP, Flyin' High or Joyride are waiting for their successors. The opportunity should not be missed. And sometimes it's better to release two or three decent games a year than ten clones or ports from 8-bit computers. That's what I wish, both for you and for my own sake. ■



AMIGA

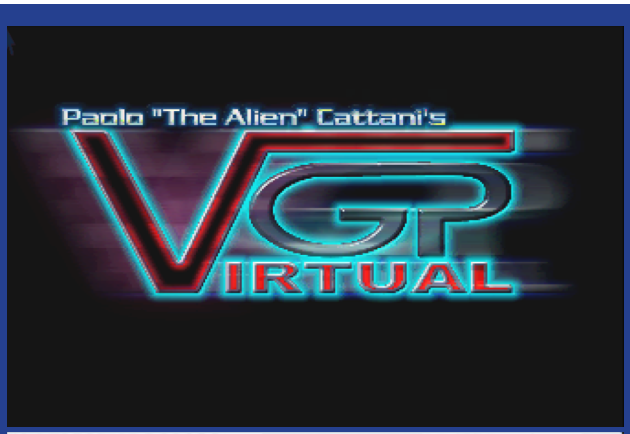
Virtual Grand Prix

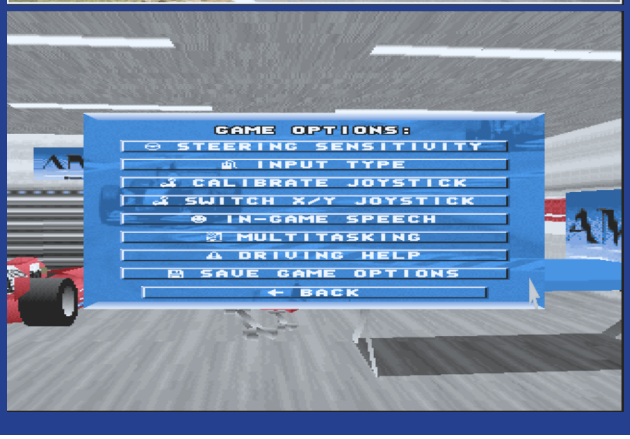
author: Don Rafito / translation: Juan

REVIEW

The roar of the engines, the smell of burning rubber and that breathtaking speed. Nothing better can happen to an F1 fan than to sit behind the wheel of a racing car. Of course, this is not always an option, but thanks to simulation games you can partially fulfill this dream. And all you need for this is... an Amiga and a game such as the one we are reviewing today.

The race of megahertz and three-dimensional graphics in the late 1990s was gaining quite a bit of momentum. The Amiga platform, despite the fact that it was by then descending into the underground, was also taking part in it. The increasingly dominant PCs and consoles were beginning to set the trends as to what a cutting edge game should look like. A man known as The Alien, otherwise known as Paolo Cattani, along with his colleagues Igor Imhoff, Oliver P. Roberts and Simon Pancri, decided to release a Formula One simulator, thus showing that the Amiga still had something to say on the subject.





After launching the title, we are greeted by a wonderful intro, showing a 3D model of a F1 racing car. Then we move on to the main menu, where we can choose Single Race or Championship mode. From here we can also load or continue a tournament. As expected, we will also find here the credits and an option to exit to the system.

In both single and multiplayer modes we can test the track in practice mode. After making our choices, we move on to the pit stop. Here we have three options: Race Car - we go to the track and start the race, Tool Box - adjusting the parameters of the car, engine or refueling, Computer - game configuration menu or exit to the main screen. From here we can also start collecting telemetry data, which will help us adjust the parameters of the vehicle to improve later results.

It's worth stopping at this point in the configuration menu, where we can adjust many parameters to our liking. These include the sensitivity of the control, the type of controller (mouse, traditional joystick and analog joystick), the speech (a voice which accompanies us while we bushwhack through the game's menu or reads to us the characteristics of a selected track), or the possibility of using helpers (i.e.: adding weight to the racing car, disabling vehicle damage during gameplay, or automatic transmission). We can run the game directly from the disk, but after installing it on the hard drive there is an option to save both the settings and our progress in the championship.

Races take place on a number of famous tracks from around the world, such as Monza in Italy, Silverstone in the UK, Spa in Belgium, Interlagos in Brazil or the Monaco circuit. The absolute minimum requirements for this title are AGA, a 030/25 processor and 4 MB of free Fast RAM, but the game will only really shine with a 68040 at 40 Mhz and a graphics card. The simulator supports full multitasking (we can switch to the system screen at any time at any time without closing it), and thanks to patches released after its release, it's possible to play in arcade mode, build our own locations and design our own racing cars.

As I mentioned in my column titled "Catching up with 3D" (which I invite you to read in this same issue), the visuals of this game could compare with those of even Playstation games at the time of its release. Not only it features a full three-dimensional engine, multicolored textures, tracks, objects, buildings, opponents' vehicles, the ability to change the view (e.g. from behind the wheel or from

behind the vehicle) but all of this available on even a vanilla AGA machine. We can run the game in several resolutions: LowRes, HighRes, Double PAL and those available for graphics cards. The only thing that might bother you is the fact that regardless of the chosen track, apart from a few details such as advertising banners or the layout of the stands (and of course the layout of the track itself), the rest of the scenery elements are very similar. Similar backgrounds, forests or skies. Unfortunately, for a Sunday racer, this may not be very attractive or even a bit tedious, giving the impression that we are constantly racing in the same location.

The pleasant and upbeat background music in the intro and a few short but decent tracks in many parts of the game's menu are basically all there is to the musical setting. To this we should add the mentioned speech in the menu (which is mostly female except when announcing the characteristics of the track) and a set of quite realistic-sounding samples that can be heard during the race itself. Not bad, although it's a pity that music is absent from the race itself. It would have been a welcome addition. Although there is a tip for that. The game itself runs its soundtrack by routing data to the Amiga's specialized chips. As mentioned, the game supports multitasking, and this means that while playing, we can go to the system at any time and run a player - be it mod, mp3 or CDDA files. The only condition is that the player in question does not block Paula for the game - by for example, using the AHI system or an additional sound card. Voila! And we have music in the game.

Virtual Gran Prix is a very good game. And it's a shame that it was at least 4 years late. Because if it had been released earlier, many Amiga gamers might have kept their friend around for longer, if only to show off this excellent game in front of fellow PC and Mac users. And despite the fact that it is quite demanding and generally not intended for casual players, it would at least statistically have found a much larger audience. Unfortunately, it hit a niche, keeping the spirits of, at best, a shrinking crowd of Amiga computer fans, while proving that you don't need "Intel Inside", a powerful graphics card and Windows 95/98 to make a good game at all.

Finally, it is worth mentioning that the author continued working on the project, releasing its second and third parts on PC. Maybe in this time, with accelerators for the Amiga classic being readily available, it would be worth thinking about porting the sequels to 68k? As gamers, readers and editors, we heartily encourage the developer to do so. ■

VIRTUAL GRAND PRIX

Developer: **Paolo Cattani/Islona**
Release year: **1999**
Platform: **Amiga AGA/GFX**

A good title, but unfortunately released at the eleventh hour - sometimes that's too late...

8 7 8

AMIGA

Paolo Cattani (The Alien)

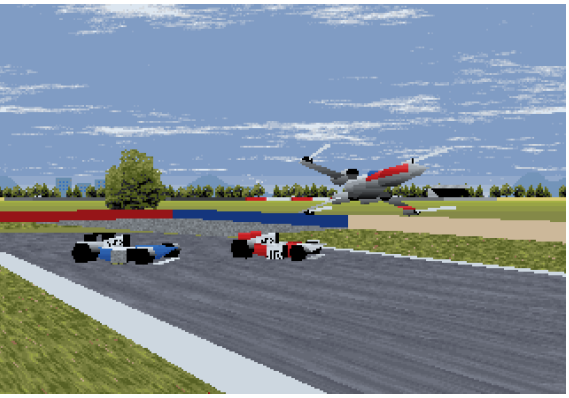
author: Juan José G. Bernabeu (Fondriesete) / translation and edits: Leon

INTERVIEW

We have the pleasure of interviewing the creator of the best F1 simulator (Virtual GP) for Amiga, Paolo Cattani. Paolo is a versatile coderman, code in Basic, C or ASM, graphic artist, musician, and simulation expert.

In your last stage with the Amiga, you wanted to make your masterpiece, Virtual GP. And what if you got it! Seeing all the technical explanations that you comment on the Web, it must have been a very ambitious project that would give you a lot of headaches. The physics of the game must also have been very complicated. Did you find it very difficult to create this game?

Quick answer: yes! VGP was, for sure, the project that required the most efforts in those days; the physics had to be realistic and accurate, the 3d engine had to be up to date with what was available on other systems (texture mapping, gouraud shading etc.), and things had to be fast on not-so-fast hardware. It was a real challenge, and yes, it gave me lots of headaches!



How much time did you invest in the total programming of the game?

Difficult to say, as I have no records of the total time spent on developing it. But I can say that it required about 3 years of development (the project started circa 1995).

Did you have any help from the F1 federation or any team to achieve the achieved realism?

No, the game was not "official" (no real names), so I got no help from the F1 federation. However, there was a friend of mine that was working as an engineer for an Italian F1 team (sorry, I cannot say more); he was so kind to show me (again, not in an official way) some invaluable telemetry that could be used to "tune" the physics of VGP.

On the year 1997 the first game of Formula 1 of Psygnosis appears on the Playstation 1, Virtual GP appears one year later for Amiga AGA. Did you want to demonstrate that with an Amiga AGA and an accelerator you could create something similar in our platform despite not having specific chips for 3D?

No, the VGP project started early, circa 1995, so it had nothing to do with the PSX F1 game.

But yes, I wanted to show that on a modern (AGA) Amiga texture mapping and other gfx tricks could be done in an efficient way. You know, Commodore had undergone bankruptcy, but in those years there were many plans to salvage the Amiga by different companies.

Many people were saying that the Amiga was dead because it was impossible to write modern 3d engines for its graphics architecture; well, I wanted to demonstrate that it wasn't the case, and that an efficient 3d engine could be developed for the AGA architecture. The idea was to show that the Amiga could still be a good gaming platform, with just a bit of upgraded CPU power, hoping that a big company could buy the whole Commodore stuff and continue the production of our beloved computer!

Once I read that it was an effort that worked that well in a 030/50. Why the other games did not go as well as this in a 030/50? It even runs on a 68020 is still playable.

It depends upon the game type, but generally speaking we can say that many 3D engines were simply "ported" from the PC world, and were not suitable for the Amiga AGA

architecture. I dont' know the reason by which many programmers did follow that way, maybe it was simply because Commodore was no more, big companies didn't invest anymore in the Amiga platform so noone had interest in researching efficient algorithms for the AGA architecture.

The Amiga was no more the "first choice", games were being developed mainly for the PC and then (somewhat) ported (if ported!) to the Amiga, so developing an engine especially designed simply for a porting was not profitable…

The 68020 cpu is 100% compatible, in terms of instruction set, with the 68030, so VGP runs on it; the real problem is that the 020 lacks the data cache, and the texture mapping algorithm of VGP is heavily based on the bit-field instructions of the 020/030 along with in-cache operations, otherwise the algorithm becomes quite slow; yes, it MAY still be somewhat playable, but not very good, indeed?

Was it a problem of the chunky2planar routines of the time or was it a problem of not knowing how to optimize the 680×0 code?

Both things, but mainly, IMHO, the fact that in many cases the starting algorithm itself was developed on a completely different hardware and couldn't be easily adapted to the Amiga architecture.

Did you use these chuncky2planar routines in the game or did you create your own routines?

The VGP 3d engine has no such a thing as a chunky2planar routine for the simple fact that the algorithm is not a "chunky pixel" one. The 3d engine of VGP works natively in the "planar space" (if I can use this metaphor) and uses directly the planar architecture of the Amiga, so no "conversion" is required. The very fact that a chunky2planar routine is required in an engine is a demonstration of what I've said above – someone is "recycling" a wrong algorithm, an algorithm developed for an "alien" hardware, so it won't, ever, achieve a very good performance (unless you have a lot of CPU power)!

(...)

There were complaints about the difficulty of controlling the car and we know that you released an update to take an arcade mode out of the game to make it easier and more affordable. Perhaps the point of view of the camera in the cockpit is too low and the track is not well appreciated?

Yes. But this is exactly how a real F1 driver sees the track, so…

Why «skid» so easily? Did you want to increase realism to the maximum?

Consider this: a Porsche 911 GT3 has 500HPs and weights about 1500KGs. Have you ever driven one? Have you ever tried to floor the gas pedal and see what happens? And, here, we are, speaking a car that has a power to weight ratio of about 1/3. Now, VGP simulates F1 cars of the 1998 season. In 1998, a F1 car had about 800HPs and weighted 600Kgs. A power to weight ratio of 4/3! Would you think it so difficult to make the car skid, with such an astonishing engine? Power on a vehicle that weights less than half a normal sport car?

Why did not you give some graphic spin effect on the wheels, so that they seem to be rolling on the track and be more realistic?

To save some CPU cycles! A (fast) rolling tyre tends to show a uniform shape, no text or logo can be read on it (everything is blurred) so the fact that they're not really rotating can be noted only when the car starts.

As for the AI of the rivals, what level of AI is in the rivals? For example, we see that they dodge or even reverse

to avoid an obstacle, how it can occur when a car is stopped in the middle of the track.

The AI in VGP is quite good, despite being somewhat rudimentary if confronted to the next chapters of the VGP "saga". If I remember well, they used a very simple algorithm (follow the idea line, brake/exit from it only if there's an obstacle in front) that was, however, quite effective.

What actions are the rivals capable of?

Well, as I said above, in the original VGP they still mainly followed a pre-computed ideal line, so that they knew where to brake and the maximum speed at which a turn can be taken (the ideal line it's always the same, so they need to adjust speed only because of fuel weight and/or tyre wear). They do so until an "obstacle" appears in the ideal line; in that case, they see if there's enough space around it to avoid it without braking, otherwise they adjust speed to change lane or come to a complete halt if no alternate route can be found. Despite being very very simple, the algorithm covers the great majority of cases (including overtaking!) that can happen on the track.

Maybe the slopes of several circuits like Monaco or Interlagos are not quite well recreated. Do you think you could have improved this?

Of course, everything can be improved, especially 3d assets! As I said above, VGP was not an official product and every piece of information had to be found by myself, including track blueprints – and some were better than others. So, it's not a surprise that some tracks were recreated in a better way than others.

The angles of the camera, the repetitions, the AI of the opponents, all this is great, and much more if we think about the time when the game came out. Do you still have the source code of the 3D engine created for Virtual GP?

I have somewhere a dump of my old Amigas' hard disks, so probably the VGP source code is still there; but, honestly, many years have passed… (...)

To finish I did not want to leave the opportunity to ask what we always ask all our interviewees, are you still using a Commodore system at present, be it Commodore 64 or Amiga?

Well, if you're referring to a REAL system (real hardware), the answer is no, I haven't turned on my old computers in many years (though I still own the C64, my first Amiga 1000, an A500+, a towerized A1200 and my last Amiga, an A4000 accelerated with a GVP 040/40). But of course I still use emulators, I have a fully working emulated Workbench 3.0 on my Hard Disk! (...)

It has been a pleasure to interview you, I do not know if you are aware that you are the creator of the best driving simulator for the Commodore Amiga, there are even people who play it with a steering wheel connected to the Amiga! (…) **We hope to see in the near future a game for Amiga created by maestro Paolo Cattani.**

Thank you very much for honoring us with your wisdom and good luck in life. Ciao! ■

The full text of the interview can be found at amigatronics.wordpress.com

Alien F1, a pre-release playable demo of Virtual GP, is available on Aminet

AMIGA

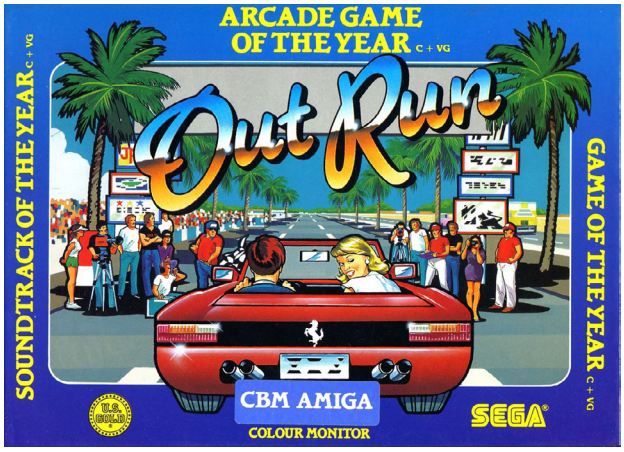
Out Run

author: Sleeva / translation: Wojtek

EDITORIAL

September 1986 is a memorable time for those who enjoy high-speed driving, the scent of burning rubber, and the deafening sound of engines that cannot be silenced by The Beach Boys' tunes blasting from the radio. This was the year when Sega brought Yu Suzuki's game, Out Run, into living rooms everywhere. Out Run was every male teenager's dream come true! A red convertible with a beautiful blonde by your side, driving along the famous coastal route.

Mr Suzuki, a legend in Japanese gamedev, aimed to create a racing game that would provide the pleasure and satisfaction of driving a high-end car. This was in contrast to the games of the time, where collisions often resulted in both vehicles exploding. Initially, he was inspired by the film The Cannonball Run and aimed to create Out Run, a game that would take players on a journey across the United States, following the same route as Burt Reynolds in the movie. However, Sega's president and Yu Suzuki's boss, Hayao Nakayama, suggested that Europe would be a safer option compared to the rather dangerous US.



Embarking on a gamedev research, Suzuki landed in Frankfurt, rented a BMW 520i, attached a camera to the roof of the car (by the way, have you noticed that sunroofs are no longer common in new cars?) and drove through Germany, Monaco, the Alps in Switzerland, Italy, the French Riviera and Spain in a fortnight. I remind you that this was the 1980s - a time when exploring and visiting real locations, as a creative process for game development, was almost unheard of. The graphics in computer games were not sophisticated enough to spend time and money travelling to specific locations without further justification. As a result, we got routes in Out Run that were a little based on Suzuki's travel experience and a little Japanese imagery of European landscapes converted to colourful pixels. However, the journey resulted in yet another element that has become a hallmark of the game. Leaving Monaco, Suzuki already knew what kind of car we would be driving in his races. On his return to Japan, he organised a photo shoot starring the Ferrari Testarossa for his team employed to produce the game, during which dozens of photos were taken of the car from every possible angle. The sound of the 48-valve V12 sports coupé was also recorded.

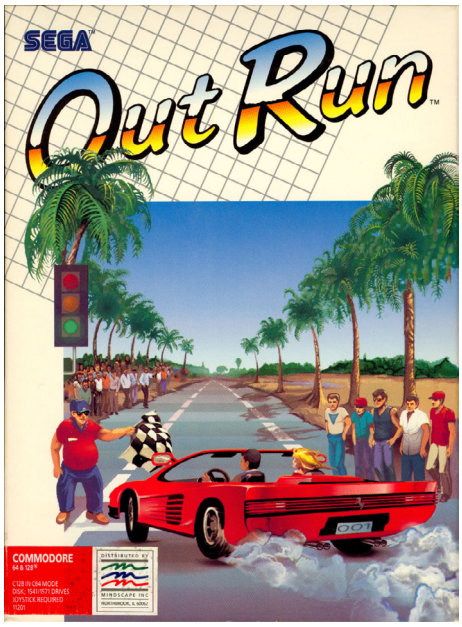
Despite the team consisting of four programmers, a sound designer, and five graphic designers, Suzuki worked after hours, designing and programming. After ten months of hard work, four versions of arcade machines were created. All four machines were equipped with a steering wheel, gear stick, and pedals for acceleration and braking. The initial two designs were vertical, standalone slot machines. One of these included force feedback in the steering wheel. However, the most intriguing models were the third and fourth ones. These weighed 350kg and were over two metres tall. They had seats that were styled to resemble a Ferrari (with a wink) and used hydraulic motors to move the entire simulator in response to the on-screen action. The sounds were emitted from two stereo speakers positioned behind the driver's head.

At the heart of those machines was a custom-built Sega OutRun motherboard based on the Sega System 16. It was decided as a best solution at a time, because Suzuki designed the games as if they were in full 3D, having nothing to do with the components in arcade machines of the time. He would plot the position of objects, scale, zoom factors and convert them back to 2D. He achieved the 3D effects in Out Run by using sprite scaling (using the Super Scaler method), which he first used in his previous game, Hang-On. This gave players a racing game that had hills and some depth, as well as trees and other objects that obstructed the view of the road and could 'zoom' in on the player very quickly.

The smooth scaling created a real sense of speed. Things almost flying right past the player. Sega's arcade hardware at the time was ahead of the curve, and Out Run was proof of the technical and artistic dominance of sprite scaling technology. In addition, Suzuki wanted to create a driving model based on the actual feel of driving. To this end, he implemented the Testarossa's real-world parameters - torque, horsepower, gear ratios and tyre grip - into the code, which provided fully satisfying control of the Ferrari including pseudo-AI-assisted drifting capabilities.

The music in the game deserves quite a separate paragraph. Firstly, Hiroshi Kawaguchi composed a soundtrack combining features of Latin music with elements of Caribbean folk, which has become a classic that is still remixed, converted and covered today. The name OutRun has been lent to an entire trend in synth music in the cate-

gory of retro-futuristic tracks from the 1980s. Secondly, three tracks: Passing Breeze, Splash Wave and Magical Sound Shower are available to choose from at the start of each stage using a screen with a car radio and the player's hand selecting radio stations, which was a recurrently used, essential element of almost every subsequent racing game (Lotus, Jaguar XJ220), and remains so today (Need for Speed, Burnout, Forza Horizon). An additional Last Wave track is played on the score screen. There is an interesting anecdote associated with the music in Out Run. Some sound samples were corrupted due to the failure of one of the main ROM chips, and the fault went unnoticed until Sega was preparing a box-set with the soundtrack for the game's 20th anniversary. The correct files were recovered from the 8-inch floppy disk and replaced in any subsequent re-release of the game.



The game itself is a typical race against time. The player controls a red Ferrari Testarossa convertible, seeing the car from a third-person perspective. The camera is placed close to the ground, simulating the Ferrari driver's position and restricting the view of anything in the distance. The road, as is common in games of this type, is quite winding, changing in elevation, making it obscured by obstacles and traffic vehicles to be avoided. The game world is divided into multiple stages, each of which ends with a checkpoint that renews the timer. At the end of each stage, the track forks, giving the player a choice of routes leading to five final destinations. The destinations represent different levels of difficulty, and each ends with its own final stage, including a Ferrari breakdown or the presentation of a trophy.

As previously mentioned, the game was released in September 1986. By November of the same year, it had already topped Game Machine's list in the vertical slot machine category and remained at the top until December 1986. It was the highest-grossing Japanese arcade game in the second half of 1986 and the seventh highest-grossing arcade game of the year. In February 1987, it became the top game on the RePlay arcade chart in North America and went on to be the most profitable arcade game of the year in the United States. Similarly, in the UK, the game held the top spot on the London Electrocoin games list from February to June 1987 and was also the best arcade game of the year.

Ports to home computers were a formality and only a matter of time. The eight-bit versions released by U.S. Gold sold over 200,000 copies within two weeks of release in the UK, and over 250,000 copies by Christmas 1987. Out Run became the fastest and best-selling computer game in the UK in the same year. By early March 1988, it had sold over 350,000 copies and became the fastest-selling game of all time in the UK. In May 1988, the Atari ST version of Out Run became the first ST title to top the UK charts. The Atari ST version sold over 25,000 copies in the UK by mid-1988.

As is often the case when porting games to hardware with less computing power, reviews frequently mention reduced graphics quality, underdeveloped and rushed conversions, or outright parodying of the original. The Amstrad version was considered one of the worst arcade game conversions ever. However, overall, the reviews were positive.

What about the Amiga? Sega's hit conversion was almost as eagerly awaited as the Atari ST version. At first glance (and, as we know, appearances are very deceiving), the updated version appears almost identical to the original, with the exception of a peculiar intro featuring a deep, artificially lowered, sampled voice and unusual music. The game now includes more detailed backgrounds and retains the three classic music tracks from the original. However, these are the only notable improvements.

The resolution of the Amiga is lower than that of the arcade machine, resulting in the Ferrari appearing slightly squashed. But that's not the worst of it. The truncated animations can be slightly nausea-inducing due to the lower frame rate causing object flickering, which can be unpleasant for some (or rather, most) viewers. Additionally, a lot of sprites were added in Amiga Out Run to enhance its visual appeal, with varying degrees of success.

The movement of the other cars is similar to that of curling championship trials, with the difference being that instead of an ice rink, we have a lawn. Overall, the speed of the action is slower than that of the Atari ST version, let alone the arcade machines.

The game Out Run is a quintessential example of how much the world has changed in the last 40 years. According to Matteo Bittanti, a professor of game design at IULM University in Milan, Out Run embodies the essence of the 80s with its fresh, laid-back, and over-the-top style. It objectifies women as a trophy for men, and holds economic and climate issues where everyone in the 80s and 90s did - in deep disregard. It's just... well, the world is rushing forward, but today's forty-somethings occasionally find themselves sitting on a seaside promenade, gazing at tanned blonde beauties in swimsuits running against the backdrop of the setting sun. And if, at that moment, a neat old-timer drives past behind you, purring with a warmed-up V12 under the bonnet, then the thought "why would anyone care..." flashes briefly through. ■

Super Sports Challenge

author: Monka / translation: Toby

EDITORIAL

September 1994. A great tram ride to the main post office for an eleven-year-old who had just taken his savings book out of the box. A saved up amount of 165k zlotys, collected from all sorts of holidays and occasions, rolled up safely in his pocket with a paper ticket he had bought much earlier so as not to waste time.

After reading the review of Super Sports Challenge in Secret Service, I was sure that, as these were the original editions of the games, the most important one would surely be in the full version and, unlike the pirated cassette copy I already owned, I would finally see what was next. In the same issue of the cult Polish gaming magazine, there was an advertisement for Mirage, the publisher of a series of Codemasters games on the Polish market.

I thought about it for a long time before I bought it. The 165,000 zloty was not a small sum for a kid. I needed a lot of time, many occasions and family events to accumulate it, and I was not one of those free-spending kids. I did the math. For that amount you could buy three more original games or four cassettes recorded in turbo, which would give you more than a hundred new titles (although the Copyright Act had only been in force in Poland for a few months, after many years of the prevalence and legality of pirated games, little had changed). On the other hand, the full version of Slicks on cassette... (this was the most important title and no one in Poland released it separately). Then there are four games that have not yet been released on turbo cassettes. Five original games in total. I remember vividly how my head was spinning.

What I don't remember is how I placed the order. Probably by letter, because there was only one telephone in the whole block, and as a minor I would not have ordered the games by telephone. There was no internet, no fast money transfers, and you had to organise everything yourself because your parents would definitely be against it. I probably sent a letter with the order, with a cash on delivery option, because I had already ordered an original Polish game that way.

I was prompted to write this column by an important event. Many of you probably got rid of your C64s in the 1990s. Some sold them for pennies, others threw them in the bin and remember that shameful act with regret. Others - like me, who was fascinated by the new technology - gave them to family members. My C64 and accessories went to my cousins, who played their favourite games with me in my youth. After a few years, I collected the unused equipment that I had intended to use for a specific purpose after the PC revolution, but I was not interested in the dozens of tapes that had been given away earlier.

After almost thirty years, I was in for a surprise - the most precious original versions of the games I had once bought came back to me. Wrapped in foil and stored, unbeknownst to me, in a barrel (!), they came to me as gifts, including Super Sports Challenge and Slicks, which I treasure with enormous fondness. The pictures of the box and the tapes you see show exactly the items I ordered from Mirage and for which I made a great trip by tram in 1994.

I was expecting the usual edition - five cassettes in separate boxes with an insert containing brief instructions. In Poland in those years, the few original editions of games for eight-bit machines were quite affordable.

When I got back from the post office and unpacked the package, I was shocked. Before I started playing, I said to myself: "It was worth it". A beautiful cardboard box with stunning colour graphics on the front, short descriptions and screenshots of the games on the back, inside a fold-out manual for all the games with the Codemasters catalogue, a large poster and finally the games themselves on three cassettes. It never occurred to me that C64 games could be released in this way. It couldn't happen because in Poland we were used to something completely different. Many games we didn't even know how to play because there were no instructions. On the other hand, when original games appeared, they were released as cheaply as possible - publishers knew that their productions would quickly find their way onto the market and sales of originals would fall.

After marvelling at the environment, it was time to launch the games, or rather one game - Slicks. The rest, as it turned out, I only tested after a few days.

Slicks is a top-down Formula 1 racing game released by Codemasters in 1992. At first glance, everything looks the same as in many other top-down racing games: small sprites of the cars, the fire button as an accelerator, left and right on the joystick... However, even though I wasn't a big fan of racing and still am not, this game suited my taste perfectly. I don't know if it was the minimalist design by the phenomenal duo of Ash & Dave (Ashley Routledge & David Saunders, Digital Design), which looked phenomenal on a 12" black and white monitor, or the brilliant music in the menus, which I played on a C64-connected hi-fi set, or the ability to challenge opponents to duels and advance to teams with faster cars, or finally the sheer dynamics of racing. It was probably all of these things that gradually influenced me to spend so much money on this game.

We start the season with the Benetton team, which doesn't have the worst car in the game, but four of the six teams have a more powerful car. So it's worth challenging an opponent with a better car as soon as possible, in order to pile up wins in subsequent races, right up to the final seat behind the wheel of a McLaren. The season consists of six races on tracks all over the world. After hundreds of races on the 'unofficial' version,

I was able to get behind the wheel of the fastest car after three races in favourable conditions, but that was as far as I went.

Only the full version allowed further battles, which, if I remember correctly, were already exceptionally easy. After dozens of hours of mastering the first three tracks, I knew exactly what speed to go at, how to shorten corners, when to ease off the throttle for half a second, and when to gently turn on the grass to overtake a faster opponent, avoiding a spectacular bounce off the pile of tyres surrounding the track. Once I had the full version, mastering the three new tracks was no problem at all, but I still played like a man possessed, and to this day I still don't know what really drew me to the screen. In 2024, my muscle memory is still intact: in a fraction of a second, I know how to enter a corner, when to fire up for the next millisecond, and which opponents to challenge to get my car into the next race. It's easy, but once I've started, I always play at least until the end of the season. Games like this are very, very rare. ■

There were four other games in Super Sports Challenge that I played with passion, after I got bored with Slicks.

CUE BOY

A billiards game that stands out for its graphics and menu animation. The very poor options, which only allowed you to play pool without being able to select its variants, made my father always choose the better versions (father, because at the age of 11 I would not have wasted time playing "balls" on the computer).

1ST DIVISION MANAGER

Football Managers were my favourite games on the C64 in the 1990s. I played all the ones that came out. This game stood out for its setting: cool music, nice cursor animation, menu graphics, game display - everything looked great. However, the most important thing was missing - even a sliver of realism of running a football club.

INTERNATIONAL SPEEDWAY

A very simple speedway game whose mastery boils down to learning when to let go of the left-hand bend so that the driver's animation ends when he exits onto the straight. I remember that I mastered this art very quickly and became the best.

WRESTLING SUPERSTARS

a computer wrestling game that looked great, but unfortunately was a bust thanks to the one punch shown in the screenshot, which only required pressing fire and swinging the joystick upwards.

New Rally-X 64

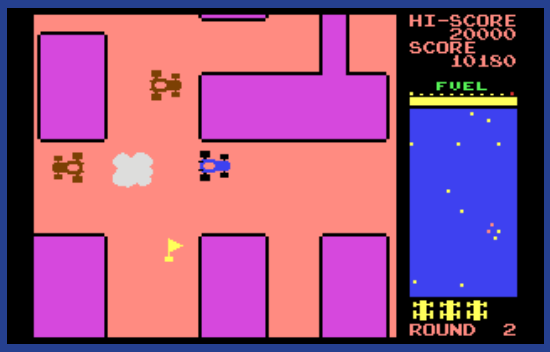
author: Komek / translation: Monka

REVIEW

Back to the past again? Yes, this time with another conversion from the first half of the 1980s. New Rally-X, as it is called, is a game released by Namco in 1981 for arcade machines, and a C64 version has just been released called New Rally-X 64.

So what do we have here? Nothing but good old fashioned fun. Although the word 'rally' is in the title of the game, it doesn't really have much to do with racing - apart from the fact that you're 'racing' with the fuel running out of your tank. The general aim of the game is to collect all the yellow flags on the board before you run out of fuel. Along the way you will encounter other vehicles that will try to stop you from completing your task. Remember, you're not completely defenceless, as you can use smoke screens to temporarily immobilise your opponents.

My dears, there's no need to go on, so I'll just end by saying that the graphics are similar to the original. As for the soundtrack, the C64 version sounds much better, but you know - SID is just better. All in all, New Rally-X64 is a successful conversion straight from the arcades to our Commodore. Oh, I forgot, there's another plus... You won't hear that chilling phrase from the older arcade regulars: "Hey, kid! Get the coins out of your pocket!. ■



The original New Rally-X64 1.1 conversion available at thestuffmade.itch.io/rally-x-64 is complete, but has the highscore saving feature disabled. Courtesy of Bacchus from the Fairlight group, the highscore function has been restored and the file can be found at the following link: csdb.dk/release/?id=219965.

NEW RALLY X-64

Publisher: **The Stuff Made**
Release year: **2022**
Platform: **C64**

May we have enough fuel to keep the machine alive

4 8 6

COMMODORE

Super GP 64

author: Erik Hooijmeijer

REVIEW

The engines roar as the light jumps to green. Off you go with the acceleration only a Formula 1 car can give, fueled by adrenaline and in pursuit of victory that borders on obsession. Will that cherished podium position become yours or will you end as a backmarker?

Super GP 64 is an improved and expanded port of last year's Super Monza GP2 for the VIC-20. There is one more track to race on, bringing the total to 5 and an in-game map now conveniently shows you your position on the circuit.

The game has two game modes with three difficulty settings: single race which lets you choose a track and you've to try and win the race and championship mode where you have to complete the race with a podium position to continue. The number of laps you choose to race determines also your starting position and the overall difficulty of the game.

Racing itself works intuitively. Keeping full throttle in a sharp corner is not a good idea and overtaking opponents requires precise timing and nerves of steel. A collision damages your car and slows you down. Damage can be repaired by making a pit stop but costs valuable seconds. Once you approach first position you also start to encounter backmarkers, the slower drivers at the rear of the race, increasing the difficulty.

There is a nice animated title screen and the 3d effect while racing is convincing. The tunnel on the Monaco circuit and the night race in Bahrain bring variety in the graphics. The screen shakes when you collide adding to the drama and the in-game map is useful for knowing when to ease down or when to floor it. The sound effects are effective, especially the swoosh sound when you pass another car (or another car passes you). The engine sound is somewhat disappointing, sounding more like a purring kitten than a ferocious tiger. Some musical accompaniment at the title and intermission screens would have added to the atmosphere in the game.

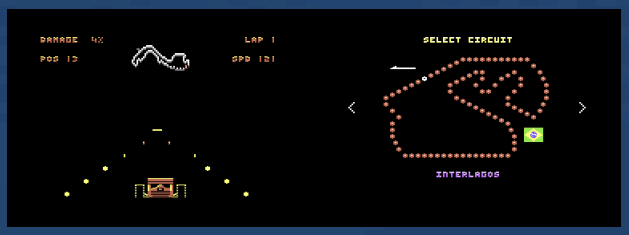
With 5 circuits, two game modes and 3 difficulty levels Super GP 64 offers plenty of racing fun. The races themselves are hard but fair and becoming champion is a real challenge. Great game. Get it at aj-layden.itch.io. ■

SUPER GP 64

Developer: **AJ Layden**
Release year: **2024**
Platform: **C64**

Verstappen, move out of my way!

7 4 7



AJ Layden

Three times race game developer?

author: Erik Hooijmeijer

INTERVIEW

Hello AJ Layden, your latest game Super GP 64 fits beautifully into the theme of this issue, but you made two racing games for the VIC-20 as well, the Super Monza GP series. Please tell us about yourself and how did your adventure with Commodore computers start?

I'm 49 and grew up in Scotland. We get a lot of rain in Scotland so staying indoors and playing computer games was a common hobby back then! My uncle had a VIC 20 and we bought it from him when he bought a C64. Likewise we then bought his C64 when he moved to the Atari ST.

So I've never been early adopter of new machines, but I often used to go to my uncle's house to play the games he had. I think Pole Position was one of the first C64 games I played. We also often played Pitstop II when that came out, so I think that period is when I really got into racing games.

After 40 years you published your first game; what re-sparked your interest?

I've always been a gamer, but never programmed much in machine language. I did learn 6502 at school on the BBC Micro but never had any access to any home development tools when I was a teenager.

After many years of working away from home, I finally decided to leave my job and move back to Scotland. I suddenly found myself with more spare time so I thought I might try my hand at programming a VIC 20 game. Rigel Attack was my first game, but I was already thinking about making a racing game while developing that.

You're now the author of 3 racing games, so why racing games? What makes them special for you?

Shoot `em ups are really my favourite genre, but racing games are a close second! I probably spent more time playing racing games with friends so I think that is what really makes them special. The VIC 20 didn't have a lot of racing games so I thought I might try making one.

What is your development setup? What tools do you use?

I develop on an old laptop using CBM prg Studio. All coding and graphics are entirely created using that. I use VICE for testing during development and then test on a real hardware once the game is nearly done.

I prioritise the features I want to include and stop when I get close to running out of memory. For me, it is not three racing games. It is really one project where each game has more features that the additional RAM lets me include.

The first game took just over three months to write. I first designed all the curves and track sections, then I worked on the scaling for the cars and trackside objects. It was all a bit of a gamble because I never saw it all moving until 2 months into the project. I had no idea if it would work or how fast it would be.

The second game had an extra 16K of RAM available so I was able to add more circuits, the pit stop and the track gradient effect. Adding the tunnel for Monaco was a bit of a pain but I needed to have it in there or I wasn't going to include the circuit.

I had no plans to write a C64 version, but one day I thought I'd see how hard it would be to take the VIC 20 code and graphics and port them over. It would give me a chance to learn about coding on the C64 without having to start a game from scratch.

The big problem is that the VIC 20 version uses a 22 wide character display and the C64 display is 40 characters wide. This is why the C64 version has a different viewpoint. It doesn't work as well as the VIC 20 version, but to recreate that viewpoint would mean starting the entire game from scratch and that was not my aim this time around. Perhaps another time!

Having managed to get the VIC 20 version up and running om the C64, I was then able to add the roadside stripes, the in-game map and improve the presentation screens a little. I also improved the opponent cars by having them occasionally change sides of the track. I worked out that I had enough memory left for another track so I chose Bahrain just to give some graphical variety.

The 3D effect works very well and is quite impressive especially on the VIC-20. How does it work and what tricks did you use to make it perform?

Arcade racers normally use a display that has shifting road perspective when you steer but this is quite CPU intensive for the poor C64 or VIC 20. Games like C64 Powerdrift and Buggy Boy have managed it but they were made by very experienced programmers!

I'm only a novice coder, so I looked at the way other VIC/C64 racing games were made. You can have fixed roads where it is centred and the player car moves. This is what VIC Pole Position or C64 Pitstop II does. However I decided to try and use the technique used by C64 games like Super Monaco Grand Prix and Out Run. This is where we have a track wider than the screen and when we steer we are moving the display window left or right, and the player car remains centred. The perspective is not entirely accurate but it is generally good enough.

The track shapes are drawn from pre-calculated pieces. I only have 5 left, 5 right and some straight values. Sixteen in total. This is why the corners might feel a bit rough compared to some games. I had to keep the number of shapes low to fit the original game into the 16K expanded VIC 20.

Because we cannot see the other side of the track when we move to one side, we would normally have a road line down the middle. I didn't have enough graphics characters available to make this so we have the road "dots". They were supposed to represent cracks in the tarmac, but I guess that needs a lot imagination!

"I've always been a gamer, but never programmed much in machine language. (...)

After many years of working away from home, I finally decided to leave my job and move back to Scotland. I suddenly found myself with more spare time so I thought I might try my hand at programming a VIC 20 game.

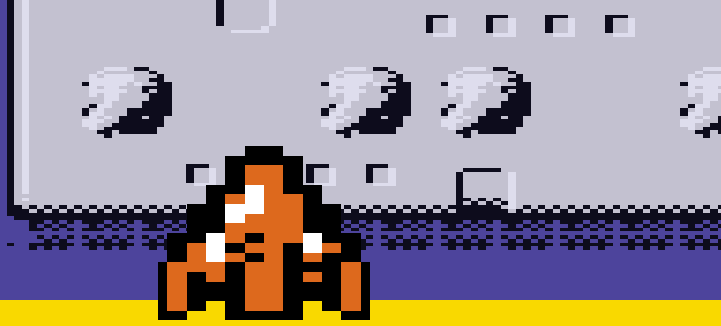
Each track section has a piece of roadside scenery associated with it. When the scenery has moved past the player then the game fetches the next piece of track and scenery. At full speed it takes about half a second for the scenery to move from the horizon to the player. So for a circuit like Monza we need about 160 pieces of track (assuming a lap time of around 80s).

What future games are you considering?

I was thinking of trying a soccer game on the VIC 20 but I don't know how well the scrolling would work until I develop it a little bit. I was also toying with the idea of a winter sports game.

I'd like to write a better race game for the C64. Start completely from scratch. ■

Corescape



author: Beszcza / translation: Toby

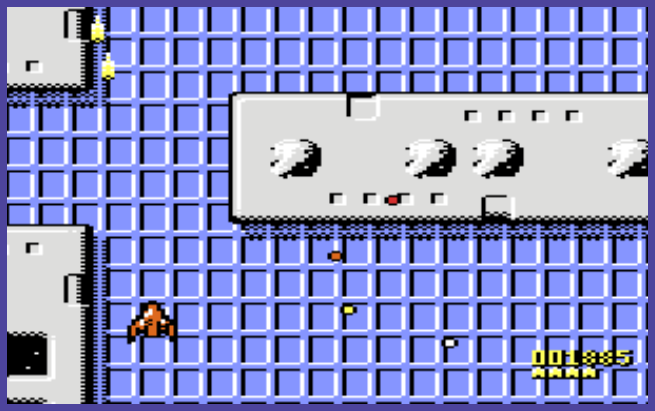
REVIEW

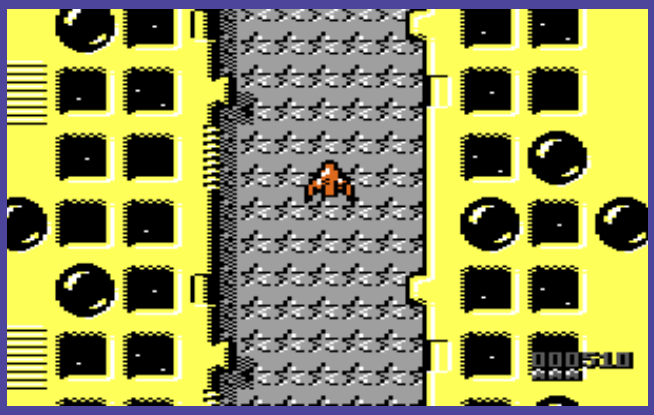
They said, "don't trust them", warned, urged to "get over it"... This was supposed to be the last shipment of corefakeium crystal and I was supposed to get over it... But it happened. I was exposed! Where are my goods and money? It's all gone! What's more, now I've got a whole empire fleet to worry about while right in the center of their space station! I knew it was not a good day for business...

Shooters are my favorite kind of games for 8 and 16 bit computers. We fly from left to right and kill everything that moves, while dodging all sorts of obstacles. I was very pleased that the Corescape game breaks this pattern and is one of the few shooters with up and down vehicle steering. In fact, the game allows us to steer our ship in any direction, additionally revealing pieces of the map on both the right and left sides. Isn't that cool? Knowing the game's author from titles such as Plekthora (shooter), Zuma (arcade), or Ball & Chain, I expected a highly polished piece of work. In the end, we received a good product, but without many extra features. Let me start with the positives. Graphic design is at a high level. You can see that it takes advantage of all the benefits of our venerable Commodore. Beautiful backgrounds, great ship explosion effects and spectacular gunfire. Add the great music and it's a hit! There's also a fair amount of enemies in the form of galactic vehicles, cannons deployed in the environment and stationary obstacles. All this should guarantee good fun, however, there is something missing.

First, the power ups, or rather the lack thereof. Where are the other weapons? Only once in many, many games did I manage to find an extra life and nothing else.

Our galactic ship takes one hit or collides with the terrain and explodes. So I ask myself: where are its shields!!!? Apparently there are bonus levels, but they only provide extra points. Speaking of points... Maybe it's an unimportant detail, but where is the highscores table with records? The game only shows the highest number of points without adding a name. Seemingly just a small detail, but I think that it would be easy to add at least the top 5. The difficulty level of the game is also super high. After a few games I felt like doing a reset and turning on R-Type. Corescape has to be learned "by heart". Very often there is only one right way and once you fly to the left, there is no way back. On top of that, I have the impression that our vehicle was slightly modified by the imperial fleet and about 25 horsepower was taken away. It takes a very long time to fly





from one side to the other, often passing millimeters away from a collectable flying gold coin.

As a consolation, the author gave us the opportunity to make the gameplay easier. Below I have included a list of facilitating keys:

T - get an infinite number of lives
L - gives additional lives
R - resets a given lvl
S - pass a given level automatically
H - slows down the game

If we want to have fun without frustration, I recommend clicking T at the beginning of the game. Of course, nothing is for free. After activating the "codes" the game will not save our score.

To sum up, the game is good, smooth, and brings a breath of fresh air to the genre, but some finishing touches are missing. Power ups would have greatly improved the game and definitely made the gameplay easier. ■

CORESCAPE

Developer: **Dr. MortalWombat**
Release year: **2024**
Platform: **C64**

And remember, shoot anything that moves!

8 8 5

Tetris Recoded

author: Phowiec / translation: Ari

REVIEW

Tetris, as it stands, can be seen by anyone. Over the years, the name has been rebranded in every possible way, so there's probably no one who doesn't know how to play it. Since 1985, when Alexei Pajytnov created Tetris, it has been mentioned in the same breath as Mario, Sonic and Pac-Man as one of the games that laid the foundations for the modern world of electronic entertainment. There's probably no console or home computer you can't stack falling blocks on.

The Commodore 64 edition of this iconic game has been redesigned to capture the spirit of the original while adding new gameplay elements. The graphics of Tetris Recoded are stunning and have a nostalgic charm that harkens back to the days when the Commodore 64 was king of the gaming world. The blocks have crisp colours and are easy to distinguish, making it easy to plan and strategise. An animation of falling blocks has been added, rather than them magically appearing at the bottom of the screen.

The controls in Tetris Recoded are intuitive and responsive, ensuring smooth and precise block stacking. The dynamism of the game is enhanced by the sounds, with a distinctive, instantly recognisable melody adding an authentic feel to the experience. From the start menu, you can choose a soundtrack to accompany your battles. There is a NES version, a Gameboy version and a brand new version. It's just a shame that the one from the original 1988 Commodore release, which Mirrorsoft was behind, isn't included.

All in all, Tetris Recoded for the Commodore 64 is a great version of a classic, combining excellent pixel graphics, intuitive controls and exciting sounds. Although it lacks a two-player mode, it offers a small substitute for competing against another human being. With WiC64 you can save your scores on the WiC64 portal. Only a substitute for online play. Despite the aforementioned drawback, the game offers an unforgettable experience for Tetris fans and retro game lovers. You should definitely give it a try! ■

TETRIS RECODED

Developer: **RetroBytes**
Release year: **2023**
Platform: **C64**

Tetris on the C64 was earlier than on the NES

6 6 8

Penguin Tower

author: Retro Chłop / translation: Ari

REVIEW

Penguin Tower is a C64 game published by Problemchild Productions. Inspired by the classic Bomberman game, it takes players into a world where blowing up walls and defeating enemies with bombs are key gameplay elements.

You take on the role of a penguin who travels through mazes full of danger, trying to find a way out of each level. Along the way you will collect treasures, keys and various bonuses to help you overcome challenges. There is also an option to play in two-player mode.

The game features aesthetically pleasing graphics, varied levels and likeable characters. Although the sounds are relatively simple, they fit in perfectly with the atmosphere of the game. Penguin Tower is an offering that will provide a lot of fun for fans of the arcade game genre. This title not only offers a dose of classic fun in a new version, but also provides exciting challenges.

Penguin Tower can be a great opportunity to refresh memories of iconic titles or simply enjoy gameplay in a proven style. ■

PENGUIN TOWER

Developer: **Problemchild Prod.**
Release year: **2023**
Platform: **C64**

Penguins playing Bomberman

5 6 5

COMMODORE

Trainyard

author: Phowiec / translation: Ari

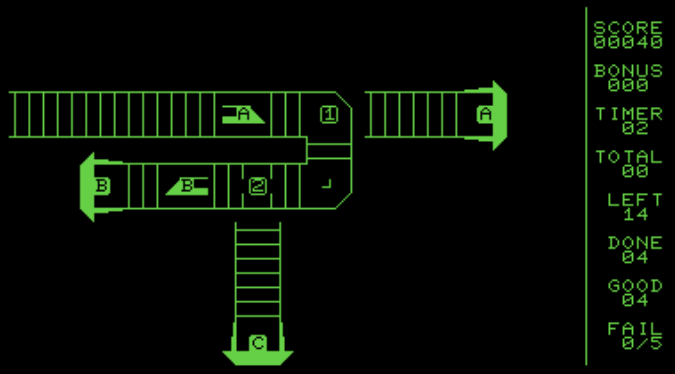
REVIEW

I wanted to be a railwayman, a real one, with a railwayman's cap and railwayman's trousers with suspenders in a big steam locomotive. When I grew up, nothing remained of my childhood fascination with steam engines, except the occasional fantasy of driving a train or controlling rail traffic. Thanks to computers, I can live out my childhood dreams for a while, even if all you see on the green screen are a few dashes and hear is a beep from the loudspeaker. Imagination is still a powerful ally, especially when it comes to running games on the Commodore PET.

Trainyard for PET proved to be a highly innovative and addictive production. The player's task is to manage railway traffic on a number of different tracks. You must coordinate the movement of the trains to avoid collisions and deliver all the loads to the required stations. As you progress, the tasks become more complex, requiring precision and quick decision-making.

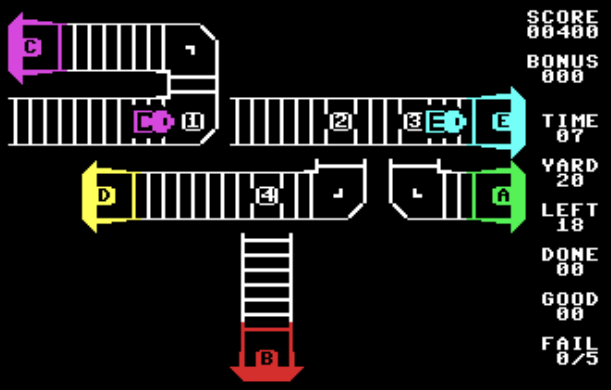
The graphics in Trainyard are what they could be on PET - simple. Despite the lack of advanced visual effects, we can feel the full satisfaction of the game.

The developers focused mainly on intelligent level design and an excellent control system, which are the game's greatest assets. The difficulty level is well reflected in the popular slogan: easy to learn, hard to master.



The lack of music can be excused by the silence, which allows you to concentrate on the gameplay - I had no expectations in this respect, as I know what kind of computer I'm using. Unfortunately, the sound of the railway switch becomes annoying after a while. I suppose there is no other way, you can always turn down the speakers to make the signalling as inaudible as possible.

In conclusion, Trainyard for Commodore PET is a unique game that delighted me with its innovative approach to logic puzzles and railway simulation. Despite its simple design, the game offers a huge dose of entertainment and challenge that will keep you glued to the screen for hours. If you're a fan of this type of game, I highly recommend giving Trainyard a chance. ■



TRAINYARD

Developer: **jimbo**
Release year: **2023**
Platform: **PET, C64**

Make the right junctions and you'll get it

5 1 8

Flappy Bird

author: Erik Hooijmeijer

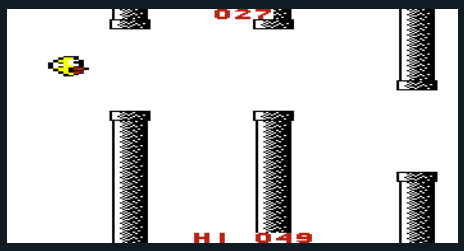
Ten years ago, the world was mesmerized by a simple game about a bird, a bird that wanted to flap around and avoid obstacles as long as possible. Now it returns for the Unexpanded VIC-20!

This is a faithful conversion of the original Flappy Bird. It's controls are simple, just press any key if you want to flap your wings and go up for a brief moment. Then gravity kicks in and sends you downwards once again. Meanwhile the screen scrolls to the left revealing pairs of pipes with gaps placed at random heights. You have to avoid crashing into them, which is frustratingly hard.

Graphically the game is quite nice, smooth scrolling, smooth movement of the bird and pixel perfect collision detection. There are some simple sound effects and the difficulty, already hard to start with, increases as you progress as the height difference between subsequent gaps increases.

One point is scored per gap passed and the high score is kept.

If you are a casual and skill obsessed gamer, this might just be the game for you. For the rest of us I'd recommend to flap away rather quickly. The game can be found on the Denial forum in the Games section. ■



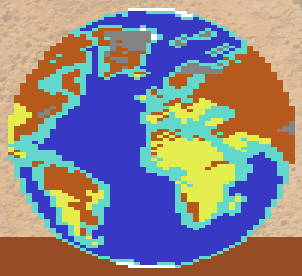
FLAPPY BIRD

Developer: **Hawks**
Release year: **2024**
Platform: **VIC-20**

Flap, flap, crash!

7 5 4

Devolution: Global Warming



author: Monka / translation: Wojtek

REVIEW

The earth became a dry and hot place. The struggle for food and water led to great, devastating conflicts. Civilisation collapsed. In its ruins, the seeds of new societies began to emerge in the form of small groups of survivors, still fighting among themselves for the remaining resources.

Devolution: Global Warming is a real-time strategy game in which our task is to unite the survivors in the wilderness. We will also have to compete with other daredevils fighting for resources.

After watching the introduction to the game, we reach the main menu. Here we can choose one of seven maps, the number of players or the difficulty level. The unusual option of choosing a colour scheme is worth noting. Depending on whether you are playing on a CRT screen or a modern LED/LCD screen, this choice can have an effect on the legibility of the small icons.

Once we have made the choices that suit us, we enter the game, which takes place on a single screen. To the left and right of it are panels that keep us informed of the state of our army, our food and scrap supplies, the items we have and the action we are about to perform. At the top of the screen is the time bar, and in the middle is the game itself.

We use the joystick to move our character icon along the marked paths between locations. When we arrive at one of the inhabited locations - a Camp, Town, Metropolis or Factory - we can choose to loot, recruit or trade. We can also attack to plant our own flag and start collecting resources. Once we have control of a location, we can fortify it and leave guards in place before moving on. Zones vary in the efficiency of loot acquisition and the speed at which people appear for recruitment. Worth noting is that only camps provide food and water, without which your people will stop following you faithfully.

In the addition to places where life is still goes on, we will also come across abandoned, uninhabitable places on the map. Some of these can be scavenged for useful items, provided they have not been taken by a competitor. Areas marked with skulls and toxic waste icons can only be traversed with a gas mask on.

The entire game takes place in real time, so it is important to remember that every action you take costs precious seconds, during which your opponents are not idle either. The journey you have started cannot be stopped before you reach your destination, so as well as planning your next moves, you will also need to think carefully about the best order in which to make time-consuming moves.

We will often encounter an enemy army on our way, but it is not possible to attack them directly. It is necessary to take over their territory and slowly cut off their resources, so that in the end they have no chance to do anything. A perfect resource denial strategy!

The game's visuals are not exactly a work of art, but the graphic designer has done a great job, and despite the plethora of pixellated icons on the board and the wealth of information, everything is surprisingly clear, allowing you to concentrate on the gameplay. On first approaching Devolution, this was not at all obvious.

The sound is limited to music, which is good but tiresome, especially after a long session on more complex boards. Particularly since it only has one track.

When I saw Devolution: Global Warming, I immediately thought that the authors were heavily inspired by the well-known Amiga game "Burntime", and I guess I wasn't far off. I really like that atmosphere and I was not disappointed. Devolution is a very good and addictive production. It gives you the "just one more turn" syndrome, which is a very good thing. It's a shame that there is no campaign mode and that the soundtrack is so underdeveloped, but maybe the authors will try to improve their great idea in the future? In the meantime, I'm having a great time playing. I'm off to conquer the wasteland once again. ■

DEVOLUTION

Developer: **Windigo Productions**
Release year: **2023**
Platform: **C64**

8-bit Burntime

6 6 8

Die Kaufmannsgilde & Spediteur

K&A+ Collector's Edition

author and translation: Tomxx

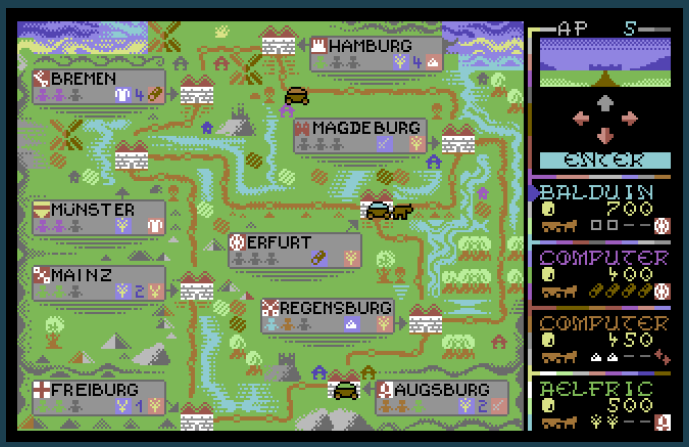
Windigo Productions, as you may have read in the review of Devolution: Global Warming and the interview on the previous pages, is the publishing brand of Friedrich Volmering. When we first contacted him as an editorial team, we didn't think it would lead to a physical release of his upcoming games. But we worked so well and efficiently together that this is exactly what happened, and today we can present you with a single collector's edition of two Windigo-branded games: Die Kaufmannsgilde and Spediteur.

Although the plot and historical background of the two games are separated by some 800 years, they both belong to the same genre - board strategy and trading games. Both are based on the game engine developed by the author and use a similar technology under the hood, e.g. path finding or multiplayer mode for up to four players (human or CPU). However, each offers its own unique set of rules and completely different objectives for players - fans of the genre will certainly enjoy the variety offered by both products.

As a publisher of physical versions, we will not be reviewing these games. This will certainly be done by other magazines or sites dedicated to C64 games. From our point of view, however, we would like to point out that it was a real pleasure to work with Friedrich, and the games you are about to play are partly the result of our feedback. We also managed to make some improvements for the Polish player: you will play Spediteur on a map of Poland and Die Kaufmannsgilde in a Polish language version.

The highlight of working on the latter game was convincing Katon to create the graphics for the title screen and the map on which the game takes place. The attention to detail in the visuals gives the game real depth and draws the player into the idealistic period of the German Middle Ages. As an added bonus, the map in Die Kaufmannsgilde is displayed in high-resolution mode, allowing Katon to pixelate many fine details that give the game a kind of painterly beauty. He worked with Friedrich himself through several rounds of feedback, trying to adapt to the developer's vision.

I invite you to take a look at our release - we have decided to release this collection on both floppy disk and cassette, as well as in the digital downloadable version! In the beautifully designed box you will also find a combined manual and postcard graphics related to the games. But most of all - you will find good and addictive games, especially for multiplayer with your friends! ■



Friedrich Volmering

Windigo Productions

author: Tomxx

INTERVIEW

Hello Friedrich, your development label - Windigo Productions - is a nice and fresh addition to the retro games community. Tell us about yourself and the reasons why you started developing games.

My family and I live in the Ruhr area in western Germany and I'm working as a lead software developer at a national IT service provider. I was born in 1980 and got my first computer, a C64, in the beginning of the 90's. I was very fascinated by computer games, but even more by how they were made. So after a short time, I taught myself BASIC only with the C64 manual and started to write my first programs. In 1992 I sent my first self-written game to a well known German disk mag publisher. Unfortunately it was rejected. Still, I was fascinated by game development. Later I experimented a little bit with assembly, but never got very far. In the mid 90s, I moved to IBM PCs where I continued writing games with my friends after school. This time mostly in C/C++ and some inline assembler. After finishing university, I worked for some years in the games industry. The C64 is still the most important system for me, for which I have so many great memories.

Can you tell us more about the environment you use?

I use a cross compiler setting with C64Studio as IDE and WinVice as runtime/debugging environment. The only other third party tool I use is DirMaster, to edit the disk images. Except for the music, all other content is created with Windigo Dreamcatcher, a tool suite I wrote. The suite contains editors for images, sprites, levels, texts, compression, asset build pipelines and much more.

On top of programming, you also create art. Is it another of your passions?

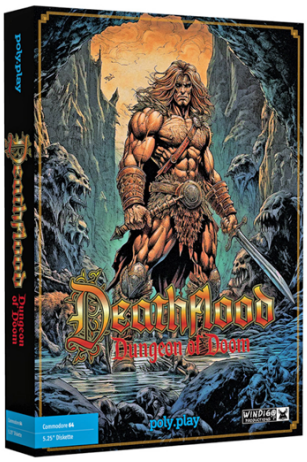
I'm not a very good artist and it takes many iterations until I'm happy with my output. But I really enjoy the process of creating pixel art. An advantage of creating graphics myself is that I can create a whole game at my own speed and without being dependent on somebody else.

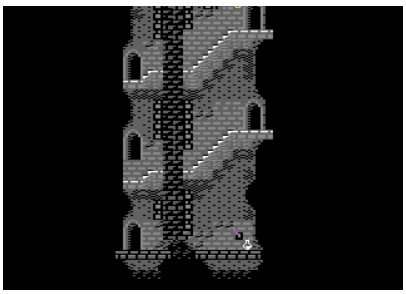
Your first published game was Deathflood - Dungeon of Doom. I know from reading your gamedev blog at www.windigo-design.de (highly recommended for curious readers) that it was heavily inspired by the Hero Quest board game and the C64's Tower of Terror. What was your experience in making this game? Have you completed all the features you wanted to do with this title?

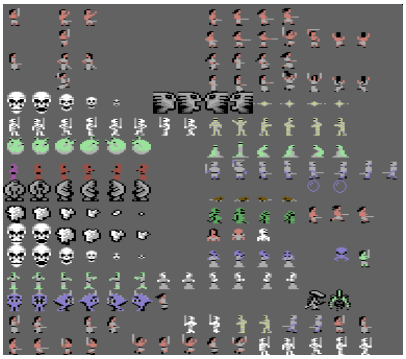
It was a long journey from the first idea until the final release. I consider myself to be quite an experienced software developer, but mastering a platform is a tough process and you need more than one project to perfect your skills. I was able to implement all features I wanted to, mostly because I developed the game in iterations: starting with the core mechanics and then adding more features and content while playtesting, until I finally ran out of memory. But this was my first full assembler project and I did have a lack of experience with the language and the platform, so if I could do it all over again, I would do a few things differently today.

The game was released by poly.play, and I wonder what the feeling is when your game is being released in a boxed collector's edition?

Holding the box in my hands was great. This was one of the reasons I did this project - I wanted to have an old school physical release of the game from the beginning. But it was quite nerve-wracking to accomplish this goal. Poly.Play wasn't the first publisher I worked with on releasing Deathflood. In the beginning I even designed a box on my own and talked to a manufacturer of board game boxes to do the production myself. But this is a lot of work and you really need a lot of experience. So I am very glad that I finally got the support from Poly.Play, who did a







great job in helping me to release a boxed version of the game.

Your next title - Devolution - is completely different as it's a real-time strategy game and you started working on the idea on a PC. How did it come about that the game was released on a C64 many years later?

I spent so much time with different versions of the project without ever being able to finish it. The main problem was that creating a classic RTS was a far too ambitious project for us when we started it in the 90's.

Just creating all the content would have required a lot more people. But there were never more than two people involved in that project at any one time. After ramping up all the tools for C64 development again, I decided to finally end this unfinished business. To accomplish that on the C64, I radically had to change the game concept, so that only the setting and a few basic ideas remained. As a result this concept change, it became possible to finish the game in less than a year and to adapt it to the C64 in terms of technical requirements and playability. After finishing the C64 version I even ported it back to PC in nearly a month, rewriting everything in C.

And the second person you worked with on this project was a musician?

That's right. The first track for the C64 version of Devolution was written by Carsten Przyluczky, who also did all the music for Deathflood. Unfortunately, he had to leave the project due to time constraints. Jörg Winterstein stepped in and created two completely new tracks. I have known both of them for years and we have worked together on various game projects. In fact, both are very talented programmers and Jörg even works full-time as a game developer. He is quite well known for his Turrican remake called Hurrican.

Your two new projects - Spediteur and Die Kaufmannsgilde - will be published in our magazine. Please tell us about their history, especially about the same core engine these projects share.

Spediteur is inspired by a German board game called 'Auf Achse' from 1987. I loved to play it as a kid and today the box sits on a shelf next to my desk. After finishing Devolution, I happened to look at the box and came up with the idea of making a project similar to this board game. When the game was about halfway done, I had some new ideas I wanted to try, like hi-res graphics. So I quickly came up with a second concept, where I could use larger parts of the technology and game rules of Spediteur, but still make it into a different game. Inspirations for Die Kaufmannsgilde are famous German business simulations for the C64, such as Hanse or Fugger. The main difference to Spediteur is that the focus is on the strategic part. The economic part is merely a supporting element. This is why there is only one map in Die Kaufmannsgilde, as in my opinion the dynamics of a match comes mainly from the voting mechanism.

I think the most important core technology both games share is the pathfinding. The distance and next step from one city to another city are stored in compressed matrices that are calculated when exporting the maps from the editor. This is a very simple approach, but allows the computer player to always find the fastest route without the need for in-game processing.

We're very grateful to you for working with our magazine and expanding your initial versions with a map of Poland in Spediteur and the Polish localisation in Die Kaufmannsgilde. Are you always so open and flexible when making games? Or do you just have a deep empathy for gamers in general?

I am also grateful for your support and feedback. That's why I'm happy to invest a little more time to respond to your ideas. Especially since, in my opinion, these are useful additions. I am always open to constructive feedback, unfortunately depending on the project status it is sometimes difficult to implement every suggestion for improvement.

After finishing these last titles, and also taking into account the previous ones, what do you find most rewarding about making games for a retro platform like the C64?

I enjoy developing for closed platforms in general and especially those with low hardware specifications because the boundaries are set and you can't solve a problem by throwing code and resources at it. You have to find creative solutions to your problems, and the limited scope of hardware makes it worth investing time in revising and improving your work.

What are your plans for the future?

At the beginning I had the plan to make exactly one game. Everything else happened that way. Although I have more or less detailed concepts for at least another dozen games, I have no idea whether I'll finish any of them. Since this is just my hobby, everything depends on the time I have available and my motivation. At the moment I'm pretty confident that I'll finish at least the second Deathflood game, as I want to try to deal with the criticism that was leveled at the first game and I've already put quite a bit of work into it.

Can you elaborate on the criticism of the mechanics of Deathflood and what are you planning to do about it? Would it be an expansion to the first game or just a whole new game as a sequel?

The main point of criticism concerns the controls. I know there is potential, but in order to do this I will have to correct some design and technical decisions, which will result in major portions of the core being rewritten. On the other hand, I conclude from the published gameplay videos that not all game mechanics are understandable, such as the rats, the time aspect or even the in-game menu. There is also a lot of room for improvement. The second part of the series will be very different, with a fresh setting and a new core gameplay mechanic.

Thank you and good luck in your further C64 development! ■

AMIGA

REVIEW

Stuntman Seymour

author: Tect / translation: Toby

Have you ever dreamed of becoming a movie star? You've probably thought more than once about what it would be like to stand in front of the camera lens, playing the same roles as the actors you admire.

Our hero, Seymour, decides to make his dreams come true and sets off to conquer the big screen in Hollywood. When he arrives, he meets director Steven Squealburg, who casts him as a stuntman in four films.

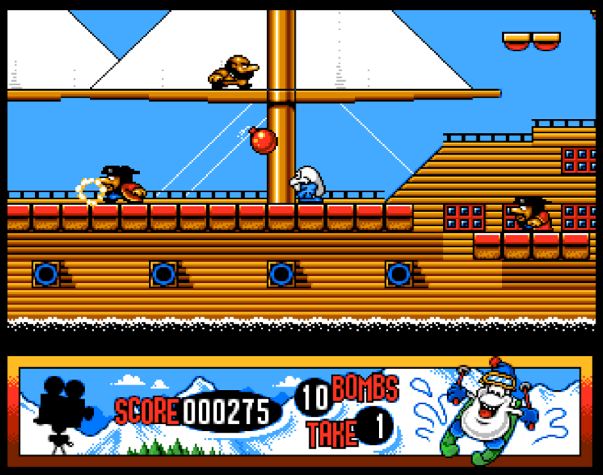
Seymour is a character best known to users of the eight-bit machines on which he made his debut. The first game in the series, called Seymour Goes to Hollywood, was created for a completely different, but well-known character called Dizzy. The developers found that they wanted to keep Dizzy in the fantasy world, so when the game was almost finished, they created a new character for him, and that's how Seymour was born. As you might have guessed, the mechanics of the first adventure game with our new hero were not too different from those of the Dizzy series. The next game maintained this style, but the following three games featuring Seymour were arcade platformers.

Stuntman Seymour was released in 1993 for the then popular 8-bit platforms. We never had the chance to play this title on the Amiga. This game was recovered from the private archives of the programmer responsible for the Amiga conversion. Unfortunately it never saw the light of day as it was deleted by Codemasters. However, it turns out that Stuntman Seymour is a fully finished and playable title on the Amiga classic.

The game debuted last year during the Christmas season, making it the perfect gift for gamers.

Seymour is a character shaped like a chubby potato with big feet and a smile on his face. On the set, he was equipped with a gun with unlimited bullets and bombs that pop out of the bodies of defeated enemies.

We begin on a film set reminiscent of the Wild West. As we jump across platforms, we come across western thugs, also equipped with guns that fire at will. There are bearded bandits in cowboy hats and Indians with axes. The enemies change according to the setting of the film, so in the next stage, nautical-themed, we have pirates throwing bombs at us or flying vultures aiming green eggs at us. Each level ends with a boss battle. After defeating him, Seymour gets into his car and drives to the next location, where the adventure of another Hollywood film begins.



The graphics follow a nice cartoon style. The smooth animation and responsive control of the protagonist are praiseworthy. The possibilities of more powerful hardware have been used to good effect. This is currently the prettiest version of the game on any platform. Unfortunately, the soundtrack is unimpressive. In addition to the title track, the same melody is played at every stage of the game. Despite the repetition, the melody never gets tiresome. This is important because it's the only thing we'll ever hear in the game due to the lack of sound effects.

Stuntman Seymour is a game from the upper echelon of platformers. If only it had been released back then, it would have joined the list of classics in its genre. Personally, I liked the game a lot and rated it higher than the Commodore version. I highly recommend this title to anyone - both Seymour fans and gamers who appreciate satisfying gameplay. ■

STUNTMAN SEYMOUR

Developer: **Codemasters**
Release year: **2023**
Platform: **Amiga OCS/ECS**

Time Capsule Game

7 3 7

Reshoot Proxima 3

author: Tect / translation: Ari

REVIEW

Here you can read about an Amiga game in the quality you know from the best arcades in the world. It's also the best Amiga game of the past year! Have I got your attention? If the answer is yes, then climb aboard, fasten your seat belt and let's fly.

In the game, we take on the role of the pilot of a spaceship called R1. Our mission is to repel the attack of an enemy alien race. Of course in this type of game, the story is of little importance, but in this case, we have some outline of the story, interspersed with spoken dialogue with the enemy during battle.

Reshoot Proxima 3 changes the rules of the game slightly. Instead of the side view used in the previous two instalments, this time a top view has been used. The first obstacles, and targets to eliminate, are meteorites, which we destroy by firing at them several times. It soon becomes apparent that these are not the only enemies we have to deal with. A whole fleet of small spaceships are heading our way. In addition, we have a wide range of enemies, with different types of enemy changing in subsequent stages.

When you are hit by a missile, the enemy vehicle is highlighted and a specific sound is heard, followed by an explosion. All of these mechanics are extremely satisfying and form the core of the game. In addition, during battle we come across ship upgrades that fly out of destroyed enemy vehicles. We can get an acceleration that improves the manoeuvrability of the ship in the sky. The second type of upgrade is a power boost, which increases the power of the vehicle. Each successive acquisition of a power boost increases firepower by redirecting it sideways, improving the range of the projectiles fired. If you are hit by an enemy, you will lose one power boost. This mechanic replaces the traditional form of life or energy bar, because if you lose all your upgrades, you are left with a basic ship - if it is hit again, the game is over.

The game requires an Amiga with an AGA chipset to run and makes good use of its potential. The graphics feature multi-level backgrounds that change during the stages. Each of them is created in a different atmosphere. We mainly fly in the atmosphere of different planets, admiring them from above. In the second level, for example, we see a blue background and animated clouds, and in the third we float above a desert overrun by a sandstorm. The animation is very smooth, considering the number of enemies and how many things are happening on the screen at the same time. The music is also great, a combination of metal and techno atmospheres reminiscent of the best demoscene productions. The sound effects, which enhance the experience, are very well chosen. We have great gunshots and explosions. The moment we pick up a ship upgrade, we hear a female voice.

The game has two difficulty modes: easy and hard. Both are challenging and it is hard even in easy mode. I recommend that you try the hard mode, in which you have to fight for a place in the high-score list, after which you will receive a special code. If you enter it at rp3high.spieleschreiber.com you will get a place on the online scoreboard, which is an unprecedented option. It adds a kind of global competition with other players and becomes an extension of the fun.

All of the above elements introduced in this title, as well as the high attention to detail, create a gameplay that has never been seen before on the Amiga. However, the amount of work and time required to make such a great game comes at a price. I bought the digital version, which includes a pdf manual, concept art and wallpapers for the workbench, which is a nice addition. The price on itch.io is 29 Euros. After currency conversion and adding VAT, the final price is PLN 161.71. At this point, you have to ask yourself if it is worth it. For me, Reshoot Proxima 3 is the best Amiga title of last year, so I didn't hesitate for a moment to buy it. If you are a fan of arcade games and especially shmup space shooters, then you know what to do. ■

RESHOOT PROXIMA 3

Developer: **Spieleschreiber**
Release year: **2023**
Platform: **Amiga AGA**

Grab the joy, there is a mission to do

9 9 8

Musical impressions

author: Hery / translation: Wojtek

EDITORIAL

I have listened to hundreds of SIDs and have encountered better and worse compositions, but none have come close to the originality and expressive power of Michał Rełkowski vel Dafa's brainchild from the Opium demo (Samar, 1997). Even among non-SID sounds, it is difficult for me to find a track that so strongly challenges my musical taste! I have never encountered such an energetic and multithreaded combination of sounds before or since. This track is a great demonstration of musical artistry and unfettered creative freedom.

It was the only track I recorded on cassette and listened to compulsively in my youth. Again and again, in the darkness of my room, I would travel to worlds created by the sounds of these extraordinary tracks. There was something magical about them. After such a powerful dose of emotion, any other sound must have been blatantly ideologically and technically inadequate.

Usually, in compositions that appeal to us, we eagerly grasp for a motif, a chorus or a phrase that evokes a sense of aesthetic satisfaction; here this is impossible, for the piece seems amorphous, with no beginning and no end - a syncretic unity. At the same time, it tells so many stories that it is easy to get lost in them.

Whereas in classical pieces we can usually find a clearly guided leitmotif without difficulty, here we are intoxicated from the outset by an artistic collage. It seems that the arrangement of the sounds does not form a well-thought-out whole, that it is chaotic, that phrases are scattered completely at random. But these are only appearances, concealing a deep structural idea. The author knows from the beginning where he is going.

The variety is reminiscent of a mix in which sounds of different dynamics and timbres are combined. What unfolds before us is a galaxy of musical miniatures, a few seconds' worth of etudes, flashing before us one after the other in a thousand images. During brief pauses, a glow of the extraordinary unfolds behind them, but we cannot fully grasp it because we are overwhelmed by a succession of feverish sensations. The whole is spiced up with numerous ornaments that subtly but significantly enrich the musical form.

Four unique tracks were used in the demo. Let's take a quick look at each of them.

Track One - The Prelude: Right from the start we get the unmistakable impression that this is an unusual composition. What catches our attention is the variety of themes that appear at a rapid pace. This variety, however, is neither tiresome nor boring; on the contrary, it is fascinating. This is the tone of the whole work.

Track two - evocative imagery: the merry-go-round speeds up! The subdued opening is enhanced by a drum beat, and cosmic sounds are heard, as if we were receiving a message from beyond! A moment later, while still gazing into the impenetrable darkness of the skies, we hear the sound of crickets chirping! Such intricate details in otherwise simple sounds! Have we gone mad?! At this point in the demo soundtrack, a flame effect appears on the dark screen. It's unclear whether this is a coincidence, but who cares! It's absolutely thrilling! But we don't have time to ponder this any longer - the train rushes on.

Track three - a storm of emotions: a quiet introduction followed by a strong, technical beat. Unpolished, rough eight-bit squeaks alternate with a sophisticated polyphonic sound. In the background, muffled sounds give the impression of an accelerated heartbeat, which is not difficult to achieve in an atmosphere of horror built on unpredictability and hardcore tones. At the end, the violent industrial cacophony gives way to a harmonious oriental sound that momentarily opens up the 'Arabian Nights'.

Track four - a total blast: a subdued beginning followed by an abrupt change in tempo and ornamentation. We are struck by an almost complete arrhythmicity, everything merging with everything else to form a unique whole. There is a remarkable charge of energy in this movement. Artistic madness reaches its peak. (On the HVSC we can also find the first version of this piece under the name "Opium (track 3 preview)"; in the conclusion it is less explosive, but also worth listening to).

While looking through Dafa's output on HVSC to find similar compositions, I noticed that among the many interesting tracks, none had the same powerful dose of emotion as Opium. This is not surprising, as achieving such a high level of aesthetic feeling and genius is rare. In an interview with Ramos (C&A Fan #3, 2008), Daf stated that: `I was fascinated by technical music and considered it more valuable (...) I was impressed by the dynamic,

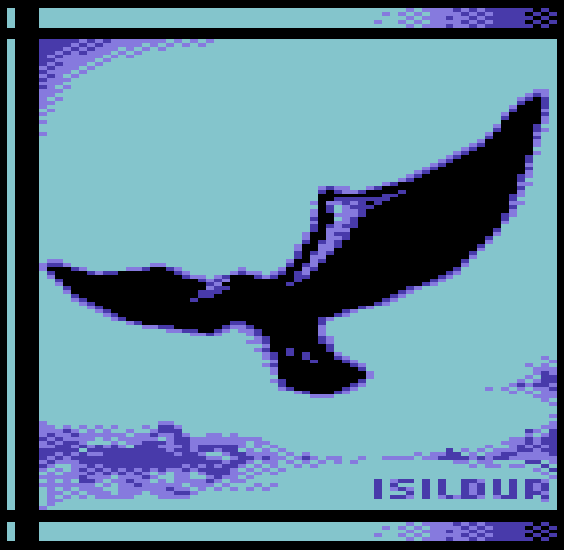
technical style that straddled the line between trance and rave.' These qualities are evident in the artist's SID portfolio. However, his earlier output seems to be only a precursor to what was achieved in Opium.

Daf is aware of this. In the aforementioned interview, he describes his creative process as follows: "I would sit down full of enthusiasm, rested, with no time pressure, and shoot something off the top of my head - those were the more successful pieces (...). I think I had the most fun writing Opium, that's when I felt the power". This "power", this flow, can be felt in every second of all four tracks. From the very first notes, we are convinced that we are dealing with a flash of genius. The piece radiates uninhibitedness, freedom and ease. The artist is in complete control of the material and expresses every idea effortlessly. As far as the C64 is concerned, this is definitely the author's opus vitae.

In this context, Dafa's words that it is difficult for him to compare himself with musicians such as Jeroen Tell, Drax or Shogoon must be seen as courtesy and exaggerated modesty. These artists have mastered other musical genres - more melodic and harmonious, evoking a nostalgic wailing of the soul. Dafa's composition is groundbreaking in that it explores entirely new dimensions of musical expression and expands our understanding of beauty.

We should not forget that the soundtrack was very closely linked to the demo. "When I wrote the music," says Daf, "I had an idea of what I was writing for (...). I wrote a simple editor that allowed me to synchronise the music with the on-screen action to within 1 frame. (...) Unfortunately, only Opium was created this way. The other demos were synchronised 'by eye', which is not the same effect". In the category 'Sound and image synchronisation', the Samar group deserves its own Oscar. Without a doubt, this is one of the best musically illustrated demos to be seen on the C64.

I could try to put into words the whole landscape of images painted by sound, the musical epic, but I would not be able to capture even a fraction of the emotions it evokes in the listener. This richness of sound from just three channels and six octaves has to be experienced. Of course, not everyone is necessarily a fan of powerful, energetic sound, but everyone will surely notice the enormity of the musical ideas born from the wild, artistic exaltation of a creatively tuned mind. ■



Party Speedway Extd. *GameDev story*

author and translation: Tomxx

EDITORIAL

Party Speedway - a pixelated racing game designed to imitate speedway racing - was developed during the two spring months of 2022. It was inspired by the game Żużel (zuzel.exe) by Piotr Kamiński, released for DOS in the 90s, which for me has always been the quintessence of unparalleled playability with an outstandingly minimalist audiovisual environment. This was also the aim of the C64 version - multiplayer, high replayability and a relatively low entry threshold. I finished the game on the same day as the Teddy Beer C=64 Party in Tuchola, where the production won first prize in the Wild category.

I could have stopped there as the game went relatively unnoticed and I didn't really promote it too widely, not even releasing it on a floppy in our magazine. But after some time, I began to think of ways to improve the game and after another year, I decided to resume work on it. My aim was to create an imitation of a riding motorbike by programmatically changing the shape of the sprite as the game progressed, adding a few more tracks that differed in shape and speed, including a competition mode and, most importantly, adding computer opponents. This last goal was a challenge for me, as creating even the most primitive 'artificial intelligence' on an 8-bit machine is not a trivial task. I worked on the project in my spare time without any time pressure until finally at the beginning of 2024 I considered the game finished and ready for release under a new title: Party Speedway Extended.

The purpose of this article is to explain some of the technical problems I encountered while programming the game and to describe the various steps I took to get it all down to a single program. I learnt a lot from this project and improved my 6502 assembler skills considerably and I am really proud of some of the solutions. I will focus on both the basic solutions implemented in version 1.0 of 2022 and the aforementioned additions to the extended version. I will try to use simple language that can be understood by non-technical readers.

THE SPRITE'S MOTION VECTOR

Initially, I didn't know how to use a single joystick button to change the sprite's rotation vector depending on its current direction of movement or how to choose the values so that the speed of movement made sense in terms of playability, easy to learn while yet being dynamic and challenging.

I solved the second of these problems using empirical methods and came to the conclusion that the offset of a sprite in a single frame should not be greater than 3 pixels. This is the combined offset of the X and Y values, so for example if X is 3 then Y must be 0, and if X = 0.55 then Y = 2.45 pixels per frame. This means that the speed at which the object moves is constant throughout the game, and turning (always to the left!) is a simultaneous change in the X and Y parameters by the same partial value. At a frame rate of 50 frames per second in PAL, this translates into a total shift of the sprite of 150 pixels in any one direction, or the corresponding number of pixels in two directions simultaneously per second. However, since the speed vector is almost always a floating point number in a single frame, the sprite can only move by the full X or Y value and the remaining partial value is added to a temporary variable and increments the vector by a full byte, i.e. 255.

Example of motion vector (2.2, 0.8) in double-byte notation: X1 = 2, X2 = 51, Y1 = 0, Y2 = 204: First frame: shift by vector (2, 0), partial values (51, 204). Second frame: shift by vector (2, 1), partial values (102, 153). Third frame: (2, 1) and (153, 102), etc. Each frame thus requires addition calculations, but the partial values of the vector make the animation appear smooth.

SINGLE BUTTON CONTROL

I solved the first problem of changing the turn vector with a single fire button by using a turn flag. For a full circle there are 4 such flags: W-S (west-south), E-S, E-N and W-N. Depending on the flag, the turn vector is either added to or subtracted from the sprite's current X and Y values. For example, the W-S flag will decrease X and increase Y, and the E-N flag will increase X and decrease Y. Depending on

the flag, reaching 0 horizontally or vertically changes it to the next flag, so the decreasing values will start to increase (and vice versa), and the sprite can complete a full loop in this way.

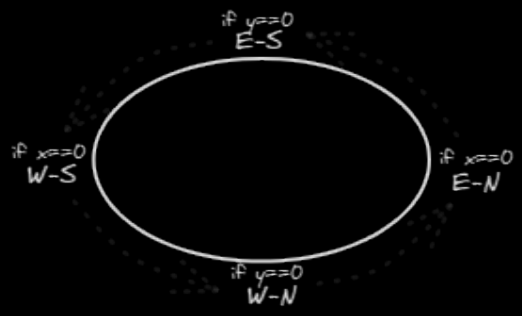


Image 1. Using flags as variables to control the direction of sprite movement

GRAPHICS AND SOUND

The main menu is divided into bitmap mode (game logo) and text mode (options, animation and scrolling text). The same applies to the race screen except that the track graphic at the top of the screen is a bitmap and the stats at the bottom are plain text. On the race screen, sprites appear on both sections - at the top are the "motorbikes" and lights, and at the bottom are the "dots" separating the times of the individual racers. The C64 has 8 hardware sprites, so the lower sprites had to be multiplexed. This was my first attempt at multiplexing sprites, which was a simple task as both screens are displayed on separate IRQ interrupts (based on the raster line) and the positions of the sprites are fixed. I could still move the stats to the bottom border as sprites (opening up the borders) and make the tracks bigger but the extra 16 pixels were probably not worth it.

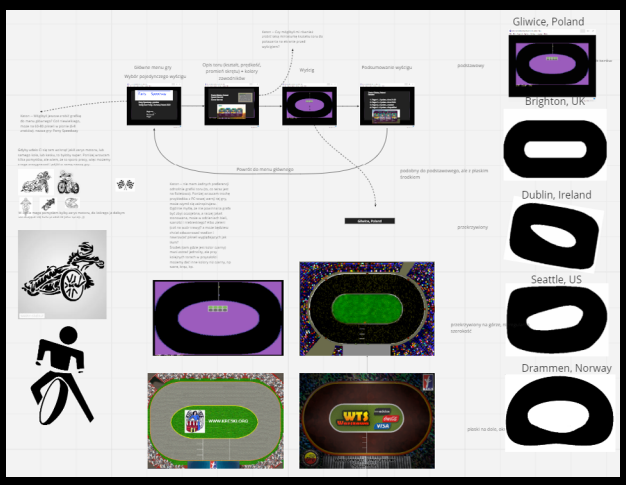


Image 2. Designing game elements on Miro.com

The graphics and music were created by my buddies from the Lepsi De group and it was the best I could ever hope for. Initially, I needed a lot of iterations and corrections (e.g. track shape settings and collision detection). Katon was patient with all my requests and implemented them immediately. We worked in a modern way, using the Miro platform where we could sketch changes and make corrections on the fly.

The music was provided by psych858o who modified his older track for me to suit the needs of the race. I needed an energetic track to emphasise the dynamics of a speedway race. Marcin cut and looped the track from his SID "Quicker". Not every artist would agree to such changes. The main menu track is a demo's track titled Attitude_21_Intro, and the sound transition to the race itself (the effect of the bikes taking off) is my creative play with the SID chip register parameters.

RELEASE OF VERSION 1.0

During the first phase of development, I concentrated solely on the race screen. Once everything related to the race - track loading, sprint control, speed, sprite's turning system, collision detection, false starts, lap counting, statistics, etc. - was working well and the race could be started and finished, it was time to add other screens. In order for the whole tech demo to be called a game, I had to tie it all together with a state manager to control the different parts of the program. This made it possible to add a pre-race screen with players ready to start, a race summary screen and a main menu with selection of race mode, number of laps and starting players. I introduced 'Best of 4' and 'Best of 8' race modes, which in turn required procedures for awarding points for each race (3, 2, 1 and 0), counting points after each round and sorting the order of players. Before each race, I also introduced a change in player readiness state when the fire button was pressed (player ready/not ready) and a 5-second countdown animation in the form of a number written in a text character, a standard-sized sprite and a stretched to double height sprite. Although I had more ideas, there wasn't enough time for them before the upcoming BOOM! Event, held in Tuchola. During this event, many people played the game and during the day I selected the best four, who in the evening competed live on the big screen. All this took place amidst a great deal of excitement and general joy of the demosceners, which gave me, as the author, a great deal of pleasure.

THE EXTENDED VERSION

After a year I decided to return to the project. In a casual conversation with DJ Gruby, I asked him how he would tackle the problem of coding computer opponents. For me, this was an important element to add value to the game and considered it to be a very interesting problem to solve, as creating computer players often exceeds the difficulty of creating a whole game.

COMPUTER PLAYERS

We quickly came to the conclusion that creating a true AI - an automaton that makes independent decisions based on input data - was definitely beyond the scope of this project. A similar effect can be achieved much more easily and without any apparent loss of quality of a computer player. Our idea was to create a predefined motion table for a full lap, depending on the route (different map shapes) and the chosen track (different Y-position when crossing the start/finish line). Admittedly, this required the creation of dozens of arrays where 1 represents a turn and 0 represents a straight line. But the calculation is done only once during programming and controlling the 'AI' during the game requires almost no further CPU processing power.

I put 'AI' in quotation marks because such rigid route coding has little to do with artificial intelligence but it is very easy to add a degree of randomness to the solution: each successive decision can be altered by introducing a pseudo-random event, i.e. a movement that is the

opposite of the one defined in the array table. Even a single substitution of a turn command for a straight command (or vice versa) will have a significant effect on the trajectory - effectively shortening or lengthening it - which can lead to an accident. The number of random events depends on the computer rider's level and the subsequent random events of improvement and deterioration in a particular round. For example, the CPU128 may run a lap perfectly (zero random events) and then suffer a significant delay or even a crash in the next lap.

However, some control was needed as with 5 random events, the path is already so unpredictable that the computer would not run a few laps successfully: so I introduced a kind of check mark, which is the start-finish line - depending on the Y-value of the sprite when crossing this line, one of 4 predefined paths is selected. This means that if a sprite crosses the line at the very top of the track and right at the edge, it will be brought to safety on the next lap using a matching predefined path.

After hundreds of trials and modifications and watching many races performed by 4 computer players, I came up with a great solution! The CPU is a tough opponent, every race is unique, and a weaker CPU can sometimes win against a better one and a better one can sometimes make a false start or run a lap much slower. This is exactly what I had in mind!

SPRITE T(R)AIL

Drawing the 'tail' of a moving object can be done in two different ways. The first is to take the position of the sprite, convert it to bitmap coordinates and place a point on it. This is not a complicated process, but it would be difficult to apply to my game as the extra pixels on the track would cause a collision like in a game of Tron. And giving up on hardware collision detection and writing it in code from scratch would be an overkill.

The second way is to programmatically change the shape of the sprite in each frame of the animation. It sounds cheesy, to be honest I'd never done it before, but in demos sprites are often modified to create, for example, scrolling text effects along a circle. Once I looked into it, it turned out that the devil is not so bad after all.

A sprite is an area of 63 bytes of memory where, in high resolution mode, each bit of a given byte lights up its specific pixel. When moving the sprite (reusing the motion flags), it is therefore sufficient to move the lit points in the appropriate direction and then add a new point at the centre of the sprite map.

Using the two built-in commands of the MOS 6502 processor - lsr (logical shift right) and asl (arithmetic shift left) - shifting bits horizontally is straightforward. Vertically, however, the corresponding memory cells must be copied manually in a loop. These are simple procedures but when shifting horizontally you have to remember to move the falling bits from one byte to the next. After coding all 4 directions, it turned out that I had to perform 63 single byte modifications either left or right (with 8 bits in it) and 63 memory copy routines of the neighbouring cells in each frame of the animation. Unfortunately, even with 3 players, there were not enough CPU cycles for this in a single frame and a nasty slowdown and stuttering occurred.

Images 3 and 4. Programmatically modified sprite traces and their view in "C64 VIC" window of Retro Debugger program

However, I managed to optimise the code (by the way, it's amazing how you can speed up 6502 code by analysing the cycles of individual commands) by using a few tricks and skipping the copying of the edge (a few first and last) bytes of the sprite. Admittedly, the 'tails' are shorter than they could be, but the animation fits within the available CPU cycles and does not stutter even with 4 players.

NEW TRACKS

A single multicolour track file in Koala weighs about 10 kB. In the memory I only have room for a single (first) track, the rest of the images have to be loaded from the floppy or cartridge. In order to reduce the loading time, I compressed the graphics and then unpack them on the fly in the program itself. I tried the popular Exomizer, but I don't really understand it and ended up using the TSCrunch cruncher (https://github.com/tonysavon/TSCrunch) written by Tony Savona for the Pig Quest game. The CPU pre-defined movement arrays are small and I keep them in memory all the time.

Similarly to version 1.0, working with Katon went very smoothly, although when he found out that the game was going to be expanded and released as a collector's edition the time it took to draw the tracks increased. He found himself demanding a certain level of quality and the tracks had to be perfectly pixelated. The effects of his work can best be seen on the Drammen track.

With the additional tracks, I changed the sprite movement speeds and turn radii. Instead of the standard 3 frames

per second in version 1.0, the speed dropped to about 2.5 frames per second on fast tracks, and even below this value in medium and slow. This helped to add variety to the tracks and to address feedback from players that the tracks were too fast and difficult to play.

Image 5. "Drammen, Norway" track by Katon/LepsiDe

ADDITIONAL MODES

With the game complete, I set about creating additional modes. I planned a time trial, a competition (tournament) and a training session where the player chooses a track and any speed. However, I had to abandon the last idea because the curves of some tracks would not work well with different turning radii.

Time Trial is a mode where a single player runs 4 laps on all 5 maps. The time of each track is recorded and a total time is calculated. A crash automatically adds 1 minute to the total time and the overall results are stored on a floppy. The main difficulty was adding up the times and sorting the results.

The Tournament, on the other hand, is a four-player competition where each track is run four times, for a total of 20 races. Players can set the number of laps before the competition, so you can play a longer game with 7 or 9 laps, or just one lap, which is similar to a precision start competition (watch out for false starts!).

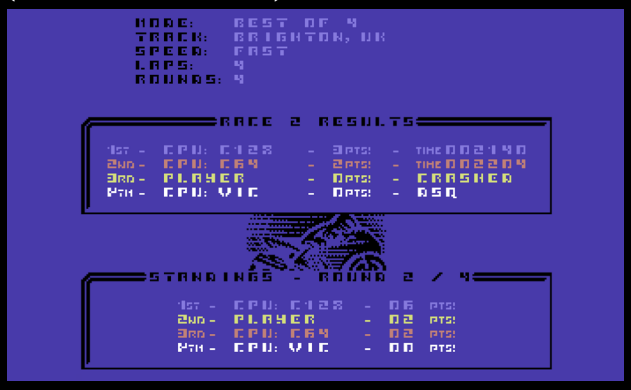


Image 6. Race summary screen

MEMORY LIMITS

And so I went on writing this code, and writing it, and writing it, until at some point I hit a wall, as I had run out of available memory. I was helped by Void, who suggested that the space above the kernal ($e000-$fffd) could be used for data, but using it required turning the kernal off and on each time by modifying memory location $1, the CPU port. This tedious process of copying data from the area above the kernal allowed me to save 8 kB of memory, which I used to store the main menu title graphics (for example). Each time the menu is displayed after a race, 2 kB of the logo image is copied to the destination memory location, which slows down the display of this screen considerably, but it is hardly noticeable, and the logo does not have to be read in from the floppy every time. It should be noted that you cannot import data into this memory area during the assembly compilation. It has to be moved programmatically from within the called program. This causes a lot of data to move in both directions in the background after the game starts, but all these operations are unnoticed by the player. Interestingly, the code required to copy such a large amount of data to the appropriate locations is stored in the screen memory area. After the first and only time it is called, the screen memory is overwritten with the display elements (e.g. the race screen) so that it is only used once. A clever solution, don't you think?

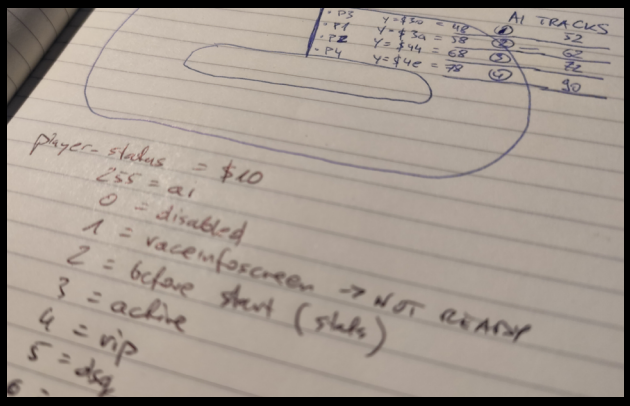


Image 7. Personal notes related to the 'AI' programming

BOXED VERSION

Party Speedway Extended lived up to its physical release! As our entire publishing house is moving towards collector's boxed editions, my game was included in this noble group. The cassette edition was never taken into account due to file loading but the game can be played on a cartridge in addition to the floppy disk. I will skip the process of creating the boxes and foam parts as this is a topic worthy of a separate article. I will just mention that I used the Midjourney bot to create the box graphics. After many attempts, the AI generated the image with a mood I was looking for - retro motes with different coloured lines to symbolise the movement of the sprites in the game. In addition to the standard floppy and cartridge sticker templates used in previous projects, Party Speedway Extended also included postcard prints with graphics from the game and a short manual to fit the dimensions of the square box. I included a few tidbits about the game in the manual, such as the fact that the cities I'd named the tracks after were places I'd lived. Just such a reminder of my travels around the world between 2006 and 2020.

As is now standard, I'm making the digital version of the game available to the retro community as a cover disk for the 25th issue of the magazine for free - and that's a fair price! - and the physical versions, including a 100-piece collector's edition signed by all three creators are available from our publisher's shop. I've also uploaded the game to itch.io where it comes with a price tag but the buyer gets a digital edition of this magazine bundled with the files, which I think is also a fair deal. ■

Lasse Öörni

Metal Coder

author i translation: Maciej „Void" Małecki

INTERVIEW

Thirty years ago, production of Commodore Business Machines' most successful product, the Commodore 64, was discontinued. The "spirit" of this little computer has survived, mostly because of its enthusiasts who still use, collect, fix and write software for the C64. Some of the productions, especially the games, have reached or even surpassed the quality we were all used to in the 80s. Maybe it's because of the convenience of cross-development, maybe it's because there's no time pressure from the industry, and maybe it's because of people like Lasse Öörni, also known as Cadaver, an author of huge titles like Metal Warrior, Hessian and Steel Ranger. Let's take the opportunity and use one of the greatest inventions called "the internet" to ask Lasse a few questions.

When I think of Finland, a few things come to mind: great vodka, metal music, Winter War, Nokia and of course the C64 demo scene. When you think of Poland, what comes to mind?

I think of the band Vader, CD Projekt Red and the Witcher franchise, several excellent newschool SID musicians like Jammer and Randall and also the past behind the Iron Curtain.

Let's focus on serious things: your opinion on the three most innovative metal bands in 2024 are... ?

I don't follow that many genres to consider myself authoritative, but if we take that to mean just the bands which I currently find inspiring, that would by Tyrantti, Beast in Black and Master Boot Record.

When did you decide to become a game developer? Was there anything that inspired you or pushed you in that direction?

I dreamed of being a game developer since being 7-8 year old and playing C64 games like Impossible Mission and Green Beret. In the early days before internet, magazines (like Mikrobitti and C-Lehti in Finland) and books were essential in pushing me on, and also the fact that even commercial games were relatively simple so that you could understand their nuts and bolts. I hopped between platforms (C64, Amiga and PC) and languages (AMOS basic, 6502 and 68000 assembly, and C) until I understood enough so that I could return to C64 making larger games with larger ambitions and modern influences, yet manageable total workload because of the C64's relatively low-res graphics & synthetized audio.

I know that you code games professionally. What programming languages do you use? Is it still good old C or is everything Pythonised?

In my day job setting most of my current coding happens in either Javascript or C# (Unity). In the past there was more of C++, it largely depends on the projects and the engine they use.

Is it difficult to compose a "metal-like" tune that sounds so good on the SID chip?

For games, I only rarely think of making an explicitly metal song (using for example power chords and palm muted riffs that an actual metal band would play). There I think the difficulty is in faster rhythm guitar and drum parts because they easily become too mechanical sounding. Anything melodic and lead-oriented (which I compose most of the time) is easier though.

Is there any C64 game release in recent years that has particularly impressed you?

I remember Knight 'n' Grail being a particular highlight years ago because it was a game that was more about being an atmospheric, indie experience rather than a "typical" C64 experience. After that we've had a steady stream of high points, like Sam's Journey, Soul Force and A Pig Quest.

When I look at the games you have written, I can clearly see that you do not take the easy way out. Multi Directional scrolling, big levels, sophisticated AI... Is there still a C64 challenge in the field of game development that Lasse would like to take on?

Some day I'd like to make an expansive top-down game, for example think of an open world Commando or Rambo where you could travel very far in any direction. And unlike BOFH: Servers Under Siege it should run at full frame rate. Let's see if that happens though.

When it comes to game coding for 8-bit platforms, is there anything you dislike, some particularly tedious coding tasks and so on?

On the C64, I kind of hate the sprites taking up CPU cycles, which means that when you're running a game with a lot of moving objects, not only the CPU logic takes time, but the sprites waste more time just by being displayed, and this leads to an ever-expanding loss of performance.

What if we run out of memory on our tiny C64? When Multiload and Exomizer are not enough, what techniques

do you use to squeeze out extra bytes?

This is a nice challenge, so I don't particularly hate it. Certainly the most important thing is to only store sprites on disk facing only one way, and decompress and flip them at runtime for displaying. Up until Hessian I thought this was too CPU-intensive to do for every sprite, but that assumption turned out false. This approach requires a "sprite cache" for the most recently displayed sprites though, so that there isn't wasted repeated work. Also, going through the game's code file by file, line by line and seeing which parts can still be shortened without sacrificing performance. At least for me it's surprising how much the initial code one writes has further potential for shortening, it can be a kilobyte or two even, out of let's say, 16-20 KB of code. For me this is also somewhere between meditation and procrastination, when I e.g. don't want to work on new graphics yet, I rather spend some time going through the code for optimization opportunities.

What hardware do you use to test your productions? Do you test on different C64 models? How about NTSC?

For daily development it's certainly just VICE. Then the Ultimate64 for testing various storage devices and also NTSC which was added as an option some time ago. Once a game would be close to finish I'd also test on an oldschool C64 + 1541 combo and a C128.

Is there a project you are working on at the moment?

Yes, I'm slowly but steadily working on a sequel to Steel Ranger. So far there's a teaser video on Youtube: (1)

I noticed that you use the multi-colour text mode - quite popular in other games, and Steel Ranger is no exception. I personally consider this mode to be the most powerful "weapon" of the Commodore 64, but I always struggle with choosing the right "ordinary" colours for the palette. Could you share some of the tips you use in your work?

At least for my graphics style I found that I should avoid using too many shades, to get a kind of easily readable and even rough quality to the graphics. For example in MW ULTRA most of the graphics is just 0 (black), 11 (dark grey), 12 (medium grey) and then the brightest color, whatever it is. You need to decide the background color so that you get the most out of single-color chars. If you use char bullets it's probably the color which makes the char bullets blend the nicest, so it's not necessarily black.

You just mentioned MW ULTRA. How do you organise your work and source code to be able to work effectively on such large and complex projects?

I work in "layers". For example in MW ULTRA, the first things to complete were the low-level scrolling and sprite display framework, then the player movement and weapons, then the basic AI of enemies. Now I had the basics of the game, but no game world yet. Next part was to draw and arrange the whole game world (as well as music soundtrack on the side). This was helped by having planned the game's story beforehand, so there was no wasted work in redoing. Now the player could walk through the whole game world, but there were no obstacles or challenge yet. So next part was drawing and implementing all the enemy types and bosses, placing them into the world, as well as placing item pickups to help the player. Now the game could be played through with appropriate challenge and combat, but there was no story or friendly NPCs or cutscenes yet. Those would be added last. This is by the way a similar sequence to how Richard Garriott worked on and finished the early Ultima games.

The code is split into the whole time resident part, and loadable code. The resident part is split according to purpose, for example inventory handling is in its own file, anything related to weapons in another etc. Loadable code which handles enemies, NPCs and story scripting, is just split into sequentially numbered files which are the loadable units, inside which are numbered entrypoints. This is not terribly intuitive perhaps, but I'm used to working this way. The game world editor references these numbered entrypoints, for example for triggering a scripted event when the player crosses a specific spot.

A few years ago you released your C64 Game Framework as open source. How much of that framework do you still use in your current productions? Do you have any plans to develop the framework further?

That framework is only used wholly in MW ULTRA, or actually it was optimized to an unreadable state when the game was pushed to completion, so the public released version represents a more general and learning-friendly version, but not as optimized. For new projects I end up rewriting large parts anyway. The loader is still almost the same, and parts of the lower level routines like scrolling remain, though improved. The public version probably won't be changed or improved, because the changes would make it completely incompatible again, and MW ULTRA represents a quite sane version. The currently worked on Steel Ranger 2 is already more specialized in its code. The source code for Hessian and Steel Ranger's free version also exist as public on github, so people can pick the one they like the best for customization.

You also write for other "retro" platforms like BOFH for PC and MW for Amiga. Which platform is the most fun to work on and why is it a C64?

I find C64 the best because of the graphics and audio capabilities are a perfect fit for me aesthetically, and also in the amount of workload. For

example, the wide multicolor pixels look very nice to me with the C64's palette, but if you were to take that palette into a 320x200 resolution, the graphics would end up looking like a Sierra adventure game with EGA graphics, which would lose the magic.

I read in some of your old interviews that you use DASM and some home-made editors for graphics and game maps. I use at least one of them myself (GoatTracker), although I am not a good musician myself (yet). Is that statement still true? What do you think about other 3rd party tools like CharPad Pro?

Yes, the statement is still exactly true. The game source releases + MW ULTRA framework include those graphics editors that go together with the game in question. For pure bitmap graphics like loading screens, I use GrafX2. For music editing, I use GoatTracker2 but convert the music to more optimized format (see (2) and (3)). CharPad and SpritePad are certainly great tools with friendly UI's, however I couldn't work on them because I always want to add additional data specific to the game. For example, MW ULTRA sprites have collision box information embedded in them, as well as points for connecting a weapon sprite etc.

I have recently started using Covert Bitops Loadersystem V3 which has saved me a lot of time and works great! But I also see that V3 is not a direct replacement for V2. Do you plan to maintain and develop both versions in parallel?

I hope that both versions are complete by now and only need bugfixes if encountered, which naturally need to be implemented separately. They have a sort of limited mission statement, and the V3 loader was already battle-tested in MW ULTRA.

Thank you very much for this interview, Lasse! We hope to play more of your games!

Thank you, it was a pleasure to answer! ■

Links used in this article

- *https://www.youtube.com/watch?v=FHxnemJsV38*
- *https://github.com/cadaver/miniplayer*
- *https://github.com/cadaver/miniplayer2*

6502 Assembly

Part II. Controlling program execution

author and translation: Maciej „Void" Małecki

TUTORIAL

A programming language is only useful if it can be used to write code that solves as wide a range of problems as possible. There is a term from computer science theory - "complete in the Turing sense" - for a language in which any algorithm can be expressed. Machine code is no different - if the language were not complete, the use of a processor that only understands that code would be very limited.

Consider the simple algorithm for multiplying two integers shown in Figure 1. At first glance, you can see that the steps required to multiply the number 5 by 4 do not follow a uniform sequence - the algorithm has branches and loops. And it is these two basic runtime controls: conditional branching (1) and looping (2) that allow us to write code that implements any algorithm.

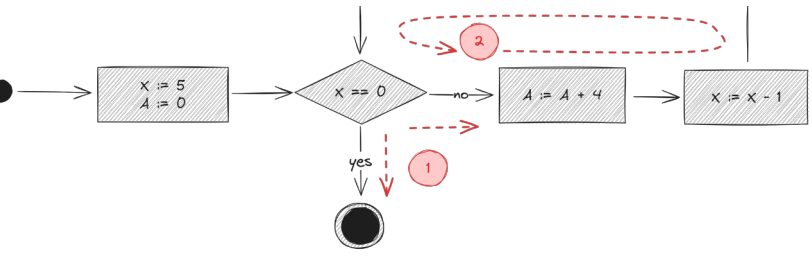


Figure 1. An algorithm to multiply two integers

How do we implement these elements in 6502 assembly language? Let's start with loops. The simplest example of a loop is an infinite loop. To implement it, you need to know the unconditional jump instruction, the most common form of which is:

```
jmp <adres>
```

where <address> is a two-byte value specifying the memory location from which the processor will begin decoding the next program instruction. The complete instruction is three bytes long, and as soon as the decoding chip recognises the instruction code 'jmp', it sends the values of the next two bytes in memory to the PCL and PCH registers respectively. If the <address> is before the "jmp" instruction, i.e. further down in memory, we can speak of a potential loop in the programme.

The simplest program consisting of a loop that displays something on the Commodore 64 screen looks like this in Kick Assembler notation (listing 1):

```
1 *= $0801 "Basic Upstart"
2 BasicUpstart(start)
3 *= $0810 "Program"
4 start:
5 loop:
6       inc $D020
7       jmp loop
```

This program increments the value of the memory cell at address $D020 by one at each iteration of the loop. This cell corresponds to the register of the VIC-2 video chip, and this register is responsible for determining the colour of the screen frame. It is worth noting that the "inc" instruction will increment the contents of memory indefinitely, i.e. the byte value will overflow. After the overflow, which will occur at 255, it will start counting down again from 0. Interestingly, the VIC-2 chip has only 16 colours, for which four bits of memory are sufficient, so in this case the overflow will occur at 15. However, this is not a problem at all if you want to cyclically repeat all the available colours - the program will work correctly despite the lack of protection against reaching extreme values.

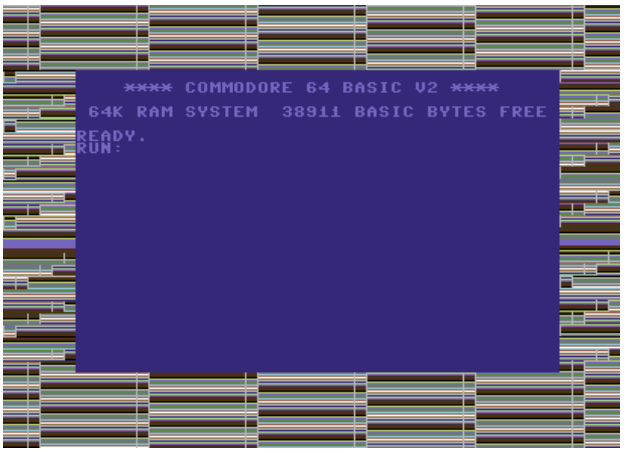


Figure 2. The result of listing 1

Figure 2 shows the effect of the above code. Interestingly, we have obtained fairly regular patterns, at least in the upper and lower parts of the image. However, at the level of the working screen, things start to look strange. Interesting, isn't it? NTSC computer owners will protest, they will see an image like the one in Figure 3.

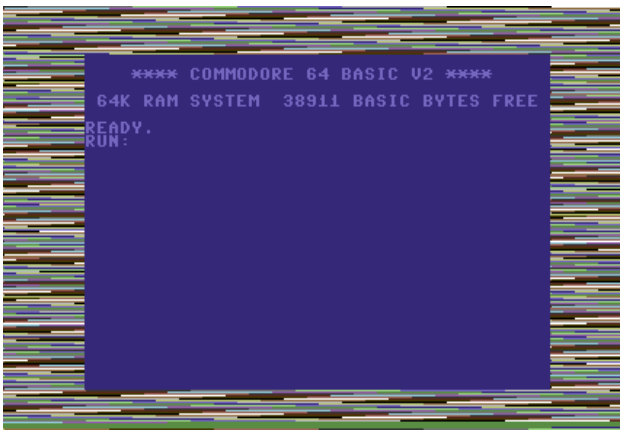


Figure 3. The result of listing 1 on an NTSC computer

The assembler language allows us to observe very subtle differences in the design of PAL and NTSC machines. In fact, it is worth noting that the "inc $D020" instruction requires 6 CPU cycles to execute, while the "jmp <address>" instruction requires 3 cycles to execute. Each iteration of our infinite loop is exactly 9 cycles. Coincidentally, it takes 63 cycles to display one line of the screen on a PAL computer, which is exactly 7 complete iterations of our loop (this explains the perfect repetition of the pattern in successive lines of the screen). NTSC computers display a line in 64 or 65 cycles, so we will not see this regularity there. The disruption of the regularity at the height of the working screen is related to the so-called bad lines effect, which I will probably write about sometime.

The "jmp" command supports two modes of addressing. The first, absolute, is the one we have just been introduced to. The second is the indirect absolute mode:

```
jmp (adres)
```

This instruction also performs an unconditional jump, but the jump target address is calculated indirectly. The value in parentheses, also a two-byte value, indicates the address from which the target address will be read. So we can easily set the location where the jump will be performed. The MOS 6502 processor has a funny bug when handling the "jmp" instruction with indirect addressing - if the lower byte of the operand contains the value $FF, then the effective address read will be incorrect.

An infinite loop is not always useful - more often we want to control the number of loop iterations performed. The Compare and Conditional Jump statements come to the rescue - we count the number of iterations and repeat the code execution as long as we have not reached the set value.

Consider the following snippet of code (listing 2):

```
1 *= $0801 "Basic Upstart"
2 BasicUpstart(start)
3 *= $0810 "Program"
4 start:
5 ldx #0
6 loop:
7       txa
8       sta $400, x
9       sta $D800, x
10      inx
11      cpx #0
12      bne loop
```

We will use the index register X as an iteration counter. This is convenient because of the existence of the "inx" instruction, which increments the contents of this register by 1. Another advantage of using an index register is a special addressing mode called direct indexed mode. Let's have a look at this mode.

When the processor encounters an instruction of the form sta $400, x (see line 8), it calculates the address to which the accumulator value is to be written in a rather specific way. To the address $400 stored as the operand of the "sta" instruction, the current value of the index register X is added and the value stored in the accumulator is written

to the address thus calculated. Many other instructions support this addressing mode, including the "lda" instruction - loading the accumulator with the value from the address calculated in a similar way.

Let's analyse our example code from beginning to end. First we initialise our counter - register X - with the value 0. This is extremely important, well-written code should never assume that registers contain specific values, in particular the value 0, before execution - we can safely assume that register values can be completely random.

Because our code writes successive counter values to two areas of memory - the first starting at address $400, the second starting at address $D800 - we use the "sta" instruction twice. The "sta" instruction writes the contents of the accumulator, but we need to transfer the contents of the X register each time, so before executing the "sta" instruction we use the one-byte "txa" instruction, which is very simple to understand: transfer the contents of the X register to the accumulator, i.e. "transfer X to A".

After writing, we increment the contents of the X register using the "inx" instruction already mentioned and then compare the value of the X register with the number 0 using the compare instruction: cpx #0. The compare instruction actually performs a simple arithmetic operation: it subtracts the contents of the register and the operand, without storing the result. The processor only stores simple facts about the last operation performed, such as whether the result obtained is zero. The "bne" operation used is a conditional jump, similar to the "jmp" instruction, except that the jump is only performed if the previous "cpx" instruction did not result in zero. The abbreviation "bne" should be understood as "Branch if Not Equal", i.e. in practice "if the resulting difference is not zero". It is also possible to reverse the jump logic by using another conditional jump instruction, "beq" ('Branch if EQual') - the jump will only be performed if the result of the last operation performed was zero.

The 6502 conditional jump instructions are very specific. All variants of these instructions are double-byte, which means that the jump address is stored as a single-byte value. In practice, this means that the target address is calculated by adding the value of the operand to the current value of the instruction counter. The addition is performed "with sign", i.e. depending on the value of the oldest bit of the operand, the jump is performed forward or backward. This has its advantages (one of which is the speed of execution of this operation compared to, for example, an unconditional jump instruction), but it also has a disadvantage: the conditional jump distance is naturally limited and, in extreme cases, it is necessary to use the "jmp" instruction to "extend" the jump.

Figure 4. The result of listing 2

Interestingly, the "inx" instruction also sets the zero flag. This happens when the value of the X register is exactly 255 just before the instruction is executed. The "cpx " instruction is therefore redundant in the case of our programme - it will be executed identically even if the instruction in line 11 is removed from the programme. It is worth noting that there are three different comparison instructions - one for each of the three built-in registers of the 6502 processor.

Let's have a look at the result of our programme (Figure 4). As it happens, address $400 contains the screen memory of the Commodore 64 at startup. Address $D800 contains the colour memory. By entering consecutive natural numbers into these two memory areas, we simply display all 256 characters one after the other, each in a different colour. Since the Commodore 64 has only 16 colours, we get a repeating colour pattern, similar to the previous example.

Let's summarise the instructions we have learnt today, which will allow us to control the programme.

COMPARISON INSTRUCTIONS	
cmp #$ff or cmp $ffff	Compare the accumulator with the argument or with the value from the indicated address.
cpx #$ff or cpx $ffff	Compare the X register with the argument or with the value from the indicated address.
cpy #$ff or cpy $ffff	Compare the Y register with the argument or with the value from the indicated address.
JUMP INSTRUCTIONS	
jmp $ffff	Make an unconditional jump to the address indicated.
jmp ($ffff)	Perform an unconditional jump to the address stored in the indicated memory cell.
bne $ff	Perform jump when a non-zero result is obtained. The jump address is calculated relatively.
beq $ff	Execute the jump when a zero result is obtained. The jump address is calculated relatively.

In the next section, we will look at what computers were made for - arithmetic. With arithmetic we will be ready to implement our first major algorithm - multiplication.

The source code for the assembly course is available on GitHub:https://github.com/ka-plus/mos6502-assembly-course. ■

How to create your own game for the C64? *Part V*

author and translation: Maciej „Void” Małecki

In the previous part of the "How to write your own game" series, I introduced you to scrolling. On a Commodore 64 computer, this technique is perfectly possible, but it is not trivial due to the very limited technical support that this computer offers. So let's start with something simple: we'll only move the screen in one direction at a time (note that many games, especially older ones, use this principle). We will also limit ourselves to only four colours on the screen.

Why is this important? In multicolour text mode, which is what we will be using, we have three colours common to the entire screen and an additional palette of eight colours individually defined for each character. If we don't use the latter, we can save a lot of work - the colour RAM used to define the individual colours for the characters can be set uniformly in this case, while a fourth colour is defined for the background, and we don't have to scroll through it. So unidirectional scrolling with a limited number of colours is the simplest background animation task we can code on our computer.

Looking through the sample CTM files using the Charpad editor, I was surprised to discover that the four-colour limitation is used in the popular 1985 game "Commando". Also, as we know, scrolling is only vertical, from top to bottom. Furthermore, the Commando game does not use any intermediate structures (tiles), so the game map can be displayed directly on the screen using relatively simple copy code.

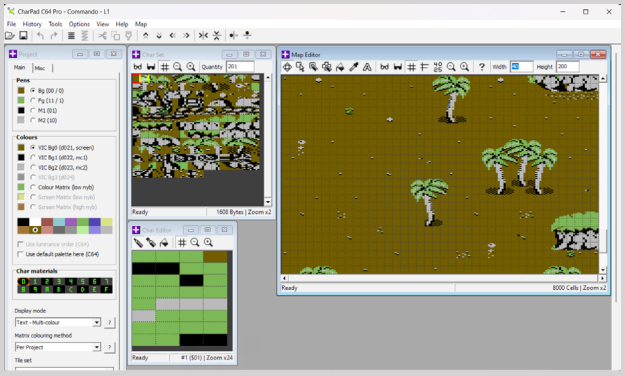


Figure 1. Level 1 of the Commando game as seen from Charpad's point of view

As I am known for my admiration of bank 3 of the VIC-II chip, we will use it in this case. For the purposes of displaying and animating the background of the Commando game, we will only need the first 4 kilobytes of this bank: two kilobytes for the character pattern definition and one kilobyte for each of the two sides of the screen memory. As I mentioned in the previous section, both sides are necessary for smooth movement, so that we do not have to copy the contents of the side currently displayed. The detailed layout of the data in memory is shown in Figure 2.

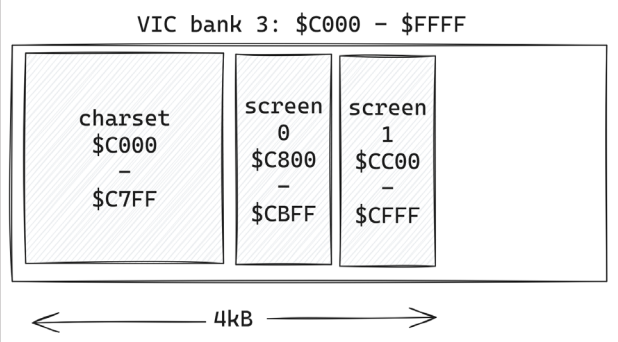


Figure 2. Graphic memory organisation

To implement the scrolling, we need a procedure to copy (or rather move) the contents of the screen, and code to fill in the missing screen line after the move. Since we have two screens and we care about speed, both the copy and fill procedures need to be written in two versions: write to screen 0 and write to screen 1. To avoid unnecessary code duplication, we will use parameterised KickAssembler language macros.

The "shiftScreen" macro copies the contents of the screen memory from the "from" address to the memory at the "to" address using an appropriate one-line shift. For further optimisation, the code generated by this macro is partially unrolled, i.e. the single-line copy is implemented in a loop, but a separate code is generated for each line (note the use of the ".for" directive).

```
.macro shiftScreen(from, to) {
    .for (var i = 24; i > 0; i--) {
            ldx #40
        loop:
            lda from + 40*(i - 1) - 1, x
            sta to + 40*i - 1, x
            dex
            bne loop
    }
    rts
}
```

So, using the macro above, we will generate two subroutines, one to copy screen 0 to screen 1 and the other to do the reverse.

```
.label SCREEN_0_MEM    = $C800
.label SCREEN_1_MEM    = $CC00
shift0to1: shiftScreen(SCREEN_0_MEM, SCREEN_1_MEM)
shift1to0: shiftScreen(SCREEN_1_MEM, SCREEN_0_MEM)
```

The macro that generates the code to display a single continuous line on the screen is a little more complicated. The data specifying the row from the level map must be taken from a memory area of many bytes (in the case of the original Commando game, this is always 200x40 or 8000 bytes). Assuming that the memory cell named "posY" contains the current vertical position of our 24-line window within the map, we need to calculate the effective start address for the data containing the row. This is a relatively simple operation involving multiplying the contents of the "posY" cell by 40 (the width of the screen) and adding it to the start address of our map. Unfortunately, such an operation is quite expensive in the 6502 machine language, as this processor does not offer multiplication operations. Assuming that the level map has been exported to a binary file (Charpad offers this possibility), we can embed it in our code and then index each row of the map using the ".lohifill" directive:

```
levelData:
        .import binary "playfield-map.bin"
levelDataEnd:

levelDataOffsets: .lohifill 200, levelData + 40*i
```

The '.lohifill' directive conveniently converts a sequence of 16-bit numbers (e.g. addresses in the 6502 processor's address space) into two arrays of bytes, each containing the low and high bytes of the number. The 'displayRow' macro code uses an index to set the base address of the copy loop. We will use the indexed addressing mode of the MOS 6502 processor, which allows us to quickly read the nth value from the byte array using the Y index register. As you can see, KickAssembler allows access to each of the two arrays containing the low and high bytes using the suffixes ".lo" and ".hi". The line is always copied to the same place in a given screen, i.e. the initial 40-byte area.

```
.macro displayRow(to) {
        ldy posY
        lda levelDataOffsets.lo, y
        sta address
        lda levelDataOffsets.hi, y
        sta address+1
        ldx #0
    loop:
        lda address:$ffff, x
        sta to, x
        inx
        cpx #40
        bne loop
    rts
}
```

We use the 'displayRow' macro to create two more subroutines.

```
displayRow0: displayRow(SCREEN_0_MEM)

displayRow1: displayRow(SCREEN_1_MEM)
```

In our program we need an additional variable to store the current status, i.e. which page is currently displayed and which one should be changed. This data could of course be stored and retrieved directly from the VIC's registers, but for the sake of code readability I prefer to use an extra memory cell for this purpose. I will call this variable 'page' and it will take only two values: 0 and 1. I will also store the contents of the vertical shift register in a separate memory cell. I will initialise all three variables in such a way that the lower part of the map is displayed at the start of the programme.

```
page:           .byte 0

posY:           .byte 176

scrollY:        .byte 7
```

Since our example code will simply display the entire map by scrolling from "bottom" to "top", we will need to periodically decrease the contents of the "posY" variable, and also modify the contents of the vertical shift register (and thus "scrollY") to achieve a smooth impression. The following procedure must be performed every screen frame, so the raster interrupt must be used. Running it at position 100 gives a nice effect reminiscent of the Commando game.

```
runEachFrame: {
        lda posY
        beq end
        inc scrollY
        lda scrollY
        and #%00000111
        sta scrollY
        lda c64lib.CONTROL_1
        and #%11111000
        ora scrollY
        sta c64lib.CONTROL_1
        lda scrollY
        cmp #7
        bne end
        lda page
        bne p1to0
    p0to1:
        jsr shift0to1
        jsr displayRow1
        lda c64lib.MEMORY_CONTROL
        and #%00001111
        ora #%00110000
        sta c64lib.MEMORY_CONTROL
        lda #1
        sta page
        dec posY
        jmp end
    p1to0:
        jsr shift1to0
        jsr displayRow0
        lda c64lib.MEMORY_CONTROL
        and #%00001111
        ora #%00100000
        sta c64lib.MEMORY_CONTROL
        lda #0
        sta page
        dec posY
    end:
        rts
}
```

This procedure first checks its execution conditions. First, it checks the condition that the scroll should already be

finished (i.e. posY is zero). Then it performs a hardware scroll and only if the position of the "scrollY" register is 7, it performs a software scroll of the screen.

Note the fragments labeled "p0to1" and "p1to0": each of these moves the screen, draws another line and switches the visible sides of the screen. Finally, the position counter is decremented and the control variable "page" is toggled.

Executing the shift takes a lot of CPU time, but it is still less than it takes to display a single screen frame, even in NTSC mode, so we do not need to divide the shift into phases. It is worth noting that this code is executed every 8th frame, so the processor still has plenty of time to execute the rest of the game logic. This is possible mainly because of the simplicity of the game (which consists mainly in limiting the number of colours). Theoretically, it would be possible to have a very high frame rate, i.e. 8 pixels per frame, but then there would be very few resources left to do anything else - and the game is not just about moving the background.

Figure 3. The appearance of the screen when the program is launched is deceptively reminiscent of the game Commando

At the beginning of this series I promised to present code that does not require access to any additional libraries. In part five, I came to the conclusion that this would not make sense. The published code would have been too long. So for today's example, I used the c64lib library set (see (1)), of which I am the author. The libraries are for the KickAssembler dialect and are available as free software (open source) on GitHub. Thanks to the libraries, I don't have to remember the register addresses of the VIC-2 chip, which makes me a lamer, but saves a lot of time. So instead of $D011 I write c64lib.CONTROL_1, instead of $D016 I write c64lib.CONTROL_2 and instead of $D018 I use c64lib.MEMORY_CONTROL.

Personally, I also don't use the Charpad tool when building software for the Commodore 64. Instead, I use a tool called RetroBuildTool (see (2)), which allows you to automate software building with scripts. This tool allows among other things the automatic export of the necessary binaries from CTM files as well as the automatic download of KickAssembler itself and the necessary libraries (in this case the mentioned c64lib). If you only have Java version 15 or higher installed on your computers, just download the example sources from (3) and, being in the root directory of that repository, execute the following command:

```
gradlew build
```

In this example, I have taken the liberty of using Rob Hubbard's brilliant score from the original game. Also, the artwork shown is from the samples included with Charpad Pro, and is actually from the original Commando game. ■

Bibliography

- **https://c64lib.github.io –** c64lib library
- **https://c64lib.github.io/gradle-retro-assembler-plugin –** RetroBuiltTool documentation
- **https://github.com/ka-plus/kammando** – the code presented in this article

COMMODORE

XC=BASIC coding guide

Part VI – Getting the final touches...

author: Tomasz „Razor" Kaniecki / translation: Juan

TUTORIAL

The previous episode of the tutorial involved the preparation of routines and variables used during the game. It is now left for us to define the behavior of the engine.

LOCKING THE SCREEN

When a block cannot move further down the screen it is locked, meaning that it becomes a permanent part of the playing field. The routine is very similar to the one in which we checked whether the block overlaps the playing field. We'll use the same function to extract individual rows from the shape, but this time we'll use the OR bit operator to merge the shape row with the game field row. Here is the function:

```
REM -- Make a piece part of the playfield
PROC lock_piece
  FOR i! = 0 TO 3
    row_no! = \piece_y! + i!
    REM -- Row 23 is the last where we want to
do this
    IF row_no! <= 23 THEN
      REM -- Get a row from the piece and merge
with playfield
      piece_row = extract_row(\shape, i!, \
piece_x!)
      \playfield[row_no!] = \playfield[row_no!] |
piece_row
    ENDIF
  NEXT
ENDPROC
```

Note that the bitwise OR operator in XC=BASIC is a vertical dash character (|).

CLEARING ROWS

Once an element is locked, we need to check if there are any complete rows to be deleted. If so, we call a routine that will delete the row and move everything downwards. We will invoke this routine for every full row.

- Again, we use a look-up table to get the memory address in screen RAM for the row.
- We then use the MEMSET instruction to clear the row.
- We also clear the row in the playfield array and then we copy the area above down row by row using MEMCPY.
- Finally, we clear the topmost row.

```
REM -- Clear a row and cascade everything from
above
CONST VICII_RASTER = $d012
PROC clear_row(row_no!)
  screen_pos = screen_address[row_no!]
  REM -- Clear row on screen
  MEMSET screen_pos, 10, 32
  REM -- Clear row in playfield
  \playfield[row_no!] = %0010000000000100
  REM -- Wait approximately half of a sec
  FOR i! = 0 to 25 : WATCH \VICII_RASTER, 255 : NEXT
  REM -- Bring everything down
  FOR row = row_no! - 1 TO 4 STEP -1
    from_pos = screen_address[row]
    targ_pos = from_pos + 40
    REM -- Bring one row down on screen
    MEMCPY from_pos, targ_pos, 10
    REM -- Bring one row down in the playfield
    \playfield[row + 1] = \playfield[row]
  NEXT
  REM -- Clear upper row
  MEMSET 1199, 10, 32

  REM -- The address in Screen RAM for each row
in the playfield
  REM -- Slightly different from as above be-
cause we only
  REM -- care about the middle 10 chars of the
stage
  DATA screen_address[] = 1039, 1079, 1119,
1159, 1199, 1239, 1279, 1319, 1359, 1399, 1439,
1479, ~
1519, 1559, 1599, 1639, 1679, 1719, 1759, 1799,
1839, 1879, 1919, 1959, 1999
ENDPROC
```

The WATCH command suspends execution until the value held by the memory address matches the required value. The address $d 012 is mapped to the VIC-II raster counter register; reading this address tells us which line on the screen is currently being updated. The WATCH $d012, 255, 255 command will wait until the register reads 255, which happens about every 1/50th of a second on PAL machines and 1/60th on NTSC machines. This command is run 25 times in a loop, which causes a wait of about half a second (slightly less on an NTSC machine).

We have written all the routines we need. We can move on to the last step, in which we will program the game loop!

CODING THE LOOP

Earlier, we designed the program flow in several loops that are nested one inside the other.

Remember the diagram in previous episodes?

- There's an inner loop, responsible for moving the current item on the game field; let's call it the update loop. Each iteration in this loop represents the downward movement of one block.
- The middle loop or game loop, which loops during

gameplay, is responsible for adding new blocks into the playing field. The loop iterates once for each new block.
- Outer loop, or the program loop. One of the iterations in this loop represents a full game.

The loop shown below updates the falling block, as long as it doesn't hit the bottom. Its algorithm is very simple:

- Draw a block.
- Check if the input was entered from the joystick.
- Check if movement is possible.
- If possible, move the block.
- Perform the above 3 times x 3 times.
- If possible, lower the block by one position or exit.

Refer to the code below:

```
REM -- The update loop
REPEAT
  REM -- Draw the shape
  CALL draw_shape(shape, piece_x!, piece_y!,
shape_color!, 1)
  REM -- The player has four chances to move
the piece
  REM -- before it falls one position
  FOR i! = 0 TO 3
    REM -- Check if there's input from joystick
    IF PEEK!(JOY_PORT1) <> 255 THEN
      REM -- Copy piece position to temp varia-
bles to be able
      REM -- to check for overlapping without
actually
      REM -- updating the piece
      tmp_x! = piece_x! : tmp_y! = piece_y! :
tmp_r! = piece_r!
      REM -- Check what input comes from joy-
stick
      IF joy_1_left!()  = 1 THEN DEC tmp_x!
      IF joy_1_right!() = 1 THEN INC tmp_x!
      IF joy_1_down!()  = 1 THEN INC tmp_y!
      IF joy_1_fire!()  = 1 THEN \tmp_r! = (\
piece_r! + 1) & %00000011
      REM -- Check if move is possible
      IF overlaps!(shape_no!, tmp_r!, tmp_x!,
tmp_y!) = 0 THEN
        REM -- It is possible, erase the piece
off of screen
        CALL draw_shape(shape, piece_x!,
piece_y!, shape_color!, 0)
        REM -- Update piece position and shape
and draw again
        piece_x! = tmp_x! : piece_y! = tmp_y! :
piece_r! = tmp_r!
        shape = get_shape(shape_no!, piece_r!)
        CALL draw_shape(shape, piece_x!,
piece_y!, shape_color!, 1)
      ENDIF
    ENDIF

FOR j! = 0 TO delay! : WATCH VICII_RASTER, 255
: NEXT
  NEXT
  REM -- Erase the shape
  CALL draw_shape(shape, piece_x!, piece_y!,
shape_color!, 0)
  REM -- Fall piece by one
  INC piece_y!
UNTIL overlaps!(shape_no!, piece_r!, piece_x!,
piece_y!) = 1
```

The code wrapped in REPEAT ... UNTIL means that this is a test loop. The loop is executed at least once, checking the exit condition at the end. If the condition is true, the loop is terminated immediately.

The CALL instruction calls the procedure.

We have not defined JOY_PORT1 and there are no such functions in XC=BASIC as joy_1_left!().
This is because they are not part of the core XC=BASIC language, but are provided by an external library or extension. Here's what you need to do before you can use the joystick function:

Download the xcb-ext-joystick.bas file from https://github.com/neilsf/xcb-ext-joystick and place it in the directory where you are working on the game.

For example, if the path to the source file is /home/mark/xcbasic-tutorial/tetris.bas, you can add the extension to /home/mark/xcbasic-tutorial/inc/xcb-ext-joystick.bas

Use the INCLUDE statement to read the extension and make it part of the program:

```
REM -- Put this line at the top of your program
INCLUDE "inc/xcb-ext-joystick.bas"
```

The path specified in the INCLUDE statement must be relative* to the current source file.

We have only implemented what we designed earlier, using calls to predefined procedures. In the next section we will test and implement the finished game. ■

Dictionary

A **relative path** is a way of specifying the location of a file or directory relative to the current working location of the file system. Unlike an absolute path, which defines the full path from the root of the file system, a relative path indicates the location of a file or directory starting from the directory you are currently in.

INCLUDE is used to enclose the contents of one source file in another source file during the compilation process. This allows you to organize your code modularly, storing definitions, procedures, functions or data in separate files, making it easier to manage large projects, reuse code and collaborate in teams.

Tapecart – Do it yourself

author: Tomasz „Razor" Kaniecki / translation: Juan

TUTORIAL

There is no need to remind our readers about how many SD2IEC-style extensions have been created for Commodore. In this short article I wanted to share my impressions of using one of them - Tapecart.

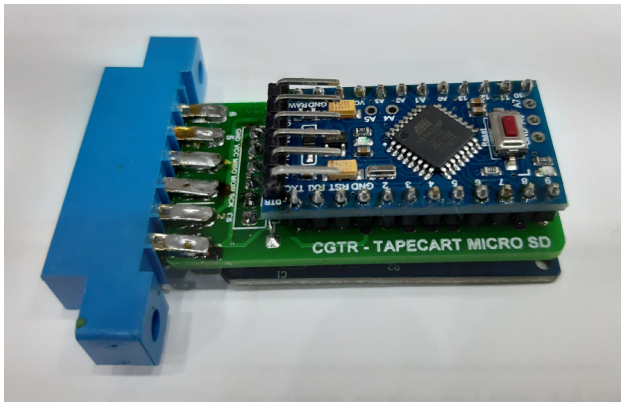
Let's start from the beginning. Tapecart is a tape recorder emulator for our Commodore. There's likely to be many of you who have been annoyed by errors while loading your favorite game due to, for example, the cassette getting old. I have also experienced this. Tapecart, a small device on whose SD card I can fit a whole world of software, looks like a convenient approach, especially since it is powered directly from the cassette port and does not require an external power supply, like the SD2IEC mentioned in the introduction.

If you feel up to it, you can assemble everything yourself, along with programming and printing the case on a 3D Printer. The board itself costs approx. PLN 10, but that's not all, because to build a complete device we need the following components, which, however, do not significantly increase the price:

Shopping cart

Arduino pro mini 5V 16 MHZ
optocoupler 4N25 or 4N27
330 Ohm 1/4 watt resistor
6 pin MICRO SD Card module
2x6 pin 3.96 mm edge connector

It is worth noting that all these components can easily be found on the Internet; all that remains is to put everything together. Here we may run into a problem if we are not acquainted with soldering and Arduino boards, but you can always consider buying a pre assembled device instead. I did just that - I bought a Tapecart from some random user off a social network to find out if this is a safe and sensible solution for my Commodore. Plug-and-play devices, complete with 3d printed packaging, cost about 25 Euro.



After ordering (hello Martin) I received a well-packaged and solid-looking "brick" in gray color (I could choose the color of the case). Tapecart requires a microSD card formatted as FAT32. In order to conveniently select prg files (this format is supported), I recommend downloading one of the wide variety of browsers available (I used browser.prg - simple to use and fast to operate). As a test, I also threw in some game, packed the hardware into an unused commodore, booted it up and... nothing. Tapecart pretends to be an actual cassette player, so there's no getting around LOAD (or more conveniently Shift+RUN). Then the magic happens. The Commodore started reading from the "tape", and after a while it found my browser and started. All smoothly and quickly. The game I had saved to the SD card was loaded without any problems.

Is Tapecart without flaws? We can say so, as long as we buy a ready-to-run hardware and aren't bothered by the somewhat rough around the edges look of PLA printed cases. On the Internet we can find hundreds of video tutorials on how to assemble and program it, so self-assembly should not be a big problem (for some it may even be extra fun).

During operation, the device did not once behave in an unpredictable manner. The prg images loaded instantly and, most importantly, without a ?LOAD ERROR, as is sometimes the case with cassettes. Neither my tape recorder nor the hundred cassettes I own will not go to the attic - I will continue to use them. Nothing can replace the fun of regulating a tape head by hand, although Tapecart will also stay around in case there's some damaged track on the tape.

Link for those interested in self-assembly: https://ccomodore64.blogspot.com/2021/11/montaz-tapecart-micro-sd-dla-commodore.html. ■

With dice and pencil

Diora

author and translation: Tomasz „RAZOR" Kaniecki

REVIEW

Before I start dissecting the board game, it would be appropriate to start with a little introduction to what is behind the title Diora. While our Polish readers should have no problem with this, a few words are in order for those who live outside of Poland.

So let's start with a quote from Wikipedia:

Diora - a manufacturing company based in Dzierżoniów that existed from 1945 to 2006, the first Polish factory to produce radio receivers in the period after World War II.

For Dzierżoniów, the company was a very important municipal institution, thanks to which people had jobs and Diora's prized products found their way to hi-fi showrooms abroad. To this day, some collectors of retro hi-fi equipment have Diora audio equipment in their collections.

Let's return to the subject of this review - the board game which was designed by Karol Stawicki and published by Dzierżoniów City Hall. The game's mechanics are what is known as worker placement - literally translated, it means 'setting up workers'. It is currently one of the most widespread mechanisms in modern board games.

We take on the role of one of the plant managers. We can choose from departments such as human resources, warehouse, production, and logistics , among others. Each manager has his own bonuses, which we can use during the game. Our goal is to produce as much audio-video equipment as possible for which we'll be rewarded with points. Our workers are k6 dices , whose number of eyes increases as we use them on the board. If any of the dices reaches the value of 6 - assigned to it worker becomes a pensioner - he can be used further in the game, but it costs an extra fee On top of regular cost such a worker - pensioner has to be paid retirement benefits every round.

The board itself depicts a manufacturing plant, with all its buildings (interestingly, it reproduces the plan exactly as it looked back in its heyday). We have a human resource building where we can hire a new employee, we have a bank that disburses the funds we need for certain operations. We have a canteen where an employee can have a meal, and we can also find a warehouse from which we take parts for production, and a production belt on which there are parts ready to be made into final product. (not sure about the last one – depending on the game board it might be better "and a production belt on which there is equipment ready for manufacturing")
To produce, for example, a radio receiver, we have to take it from the belt (-1 worker) and then pay for it with the appropriate components (e.g. 3x cable, 2x speaker, 1x transistor).

The game goes on until we exhaust all the event cards (we uncover them every round). The game is made well, the cards are legible, and the board is adequately rigid. The graphic design by Michal Stawicki aptly refers to the era, it is slightly cartoonish. The appearance of the equipment is also very well rendered. The rules are simple and can be quickly grasped, so even younger players should have no problem understanding them. The game works well for two - three players, although you can easily play solo for practice (there are no rules for playing alone, but you can safely define your own, such as those for blocking certain fields).

Currently the game is published only in Polish, but who knows, perhaps the author will release a pdf with an English rule book? After finishing a game, one is already thinking about the next one - and that's what board games should be about. With clear conscience I can recommend "Diore" not only to aficionados-audiophiles, but also to those who dreamed of being a director of a large factory. ■

On another subject

authors: Drakon, Komek, Razor / translations: Toby, Wojtek, Razor

REVIEW

Alloy Box

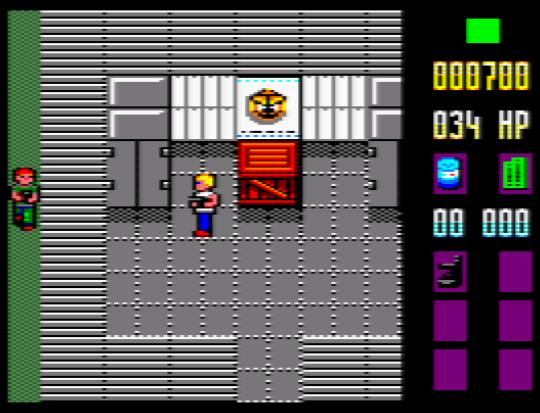
Here's a nice treat for fans of Metal Gear-style games. A new game for the Amstrad CPC has recently been released called Alloy Box, which refers to the hit game from years ago. So get acquainted with the details that I will present to you below.

The protagonist of the game is a soldier who has to infiltrate a military area and then use the dynamite he has acquired to destroy a safe in order to retrieve the great secret hidden inside. At the start of the game, the first thing to do is to try and get some weapons, as your bare fists will not be enough in the long run. Magnetic cards also play an important role, allowing you to open blocked passages.

All the necessary items are hidden in green boxes marked with a question mark. The game is a very large maze crawling with soldiers with rifles and grenades. Well, nobody said it would be easy.

All in all, Alloy Box is a nicely done production. The colourful (as befits an Amstrad) and interestingly pixelated graphics are impressive, and the music and sound effects are quite good. As for playability, it is at a high level thanks to the fast, fluid and ever-changing action. I recommend the game to all gamers, even those who are not necessarily fans of the genre. ■ **Komek**

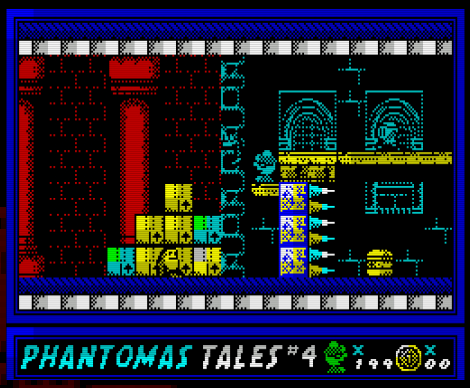
Platform: Alloy Box, Amstrad CPC, cngsoft, 2023



Phantomas Tales #4: Severin Sewers

Probably every retro platformer fan knows (or at least should know) the Mojon Twins group. The authors of titles such as Zombi Mall, Sir Ababol, Uwol and Lala Prologue have already managed to get us used to the high quality of their productions. Phantomas Tales is no exception. Seemingly an ordinary platformer, an ordinary arcade, yet with just the first impression of the game you can see that everything has been done with impressive attention to detail.

In Phantomas Tales 4, we control... Phantomas, whose goal is to collect 15 gold coins hidden throughout Metro City and the network of sewers that run beneath it. This is pretty much the end of the plot description (well, because why should I write about "mysterious acolytes of the Vehjstiahows sect"? Nobody will understand it anyway).



What makes Uwol, sorry, Phantomas such an appealing title is its visuals, which are attractive despite the omnipresent colour clash. The dynamic gameplay is complemented by some pretty good music that fits the atmosphere of the game perfectly. An interesting graphical solution has been used to represent the darkness in the sewers beneath Metro City: only a fragment of the map around the protagonist is displayed on the screen (unfortunately, I don't know what the source of light is - the protagonist himself, or perhaps something else). This intensifies the player's concentration because you can't see what's right next to you. We only have one life, while every time we come into contact with an enemy or after contact with sewage, we lose energy, which can be regenerated by eating food left here and there. The controls are also very interesting. There are two types of jumps in the game - high and wide. Play and you will find this solution very practical.

Originally, the game Phantomas Tales #4: Severin Severs was created to commemorate the 25th anniversary of the release of the first Phantomas on the ZX Spectrum (1986), while the game I am describing, released in 2024, is its remaster.

You can find the game at: https://www.mojontwins.com/juegos_mojonos/phantomas-tales-4-severin-sewers. ■ **Drakon**

Platform: *ZX Spectrum 128k, The Mojon Twins, 2011, 2024*

Sorcerer Kid Adventure

The market for games for the ZX Spectrum is certainly not one to complain about. Every month, developers send us at least a couple of titles for the platform. Of course, this quantity is not always accompanied by quality....

Sorcerer Kid Adventure is a typical example of the platform arcade genre, of which there are many on the ZX Spectrum. The plot seems promising: "Young Argos is trying to free his people from the tyranny of Master Ghraul, who has ruled them for years through fear, aided by creatures from the depths of hell. Legends say that those who learn the spells contained in the Holy Book will gain such magical power that they will be able to face the most hideous creatures of the underworld. Some consider this a childish tale, but Argos, aware of the legend, undertook the endeavor to face countless dangers to reach the heart of the Dark Castle and learn the spells contained in the Holy Book". Sounds pretty intriguing, doesn't it? Well... The plot is perhaps one of the game's few strengths. In this game, we control Argos, the goal is clearly defined, and in the course of our search for the three fragments of the Holy Book, we will encounter a large number of enemies who will try to hinder us in this quest. However, it's not the enemies that are the most difficult part of the game, but... the difficulty caused by the clumsy controls. Unfortunately, even most of the movements we make, like jumping from platform to platform, have to be done 'pixel-perfect', as the game does not forgive even the smallest mistakes. At the beginning we have 5 lives, but on the second screen you will see that this is far too few. On the second, because on the first .... there are no opponents. This game is a little bit like a souls game - some people will be put off by this level of difficulty, but others will love it, because it is this feature that makes you even more curious about what you will find on the next screen. Later on, of course, you can get magic weapons, but even these are not effective against all types of enemies. I noticed some music, but honestly you can just turn it off. I recommend the game, but with caution, because you can destroy the joystick/keyboard.

Sorcerer Kid Adventure can be downloaded from: vidaextraretro.itch.io/sorcerer-kid. ■ Komek

Platform: ZX Spectrum, Vidaextraretro, 2024

A journey to the land of wood and coal - Atari 2600+

I can already imagine a scene straight out of Frankenstein as you, dear readers, descend on me with torches and pitchforks because the worst enemy of every Commodore fan has appeared in the pages of our magazine - Atari. But is that still the case? Do we still argue about which hardware is better? Isn't there an Atari XE on your desk or in a display cabinet next to a Commodore 64?

The console that was my first "Video Computer System" was the Atari 2600. In fact, I happened to have something called "Rambo TV Game", which was just a knock-off with dodgy joysticks, but it looked and worked like the Atari in question. It also had 128 games built in.

So how could I resist buying one when I saw a colourful box with the words "Atari 2600+" on it?

Interestingly, this is an official Atari product (yes, a true daredevil was found, that bought back the rights to the brand and started making money on retro). The console looks exactly like its great-grandfather but is slightly smaller. Inside you won't find the famous MOS 6507 processor clocked at 1.19MHz, but a modern Rockchip 3128 processor with 256MB of DDR3 RAM and 256MB of built-in eMMC memory.

When you boot up the console, the Atari logo appears on the screen, and the case also lights up with a cool LED logo on the front, which was not present on the original version. A great idea are the two DB9 ports to which you can connect any joystick from the era (a single copy of a famous CX40 manipulator is also included). Any power supply with a USB-C connector and at least 1A will work with the 2600+ (not included in the kit).

The console is connected to the TV via an HDMI cable, so it runs seamlessly on any modern screen. Admittedly, the square graphics of the original games running on 60 inches are a little intimidating, but what do you expect?

Speaking of games, a 10 in 1 cartridge is included - and here's a surprise, most of the original cartridges fit the console (a list of compatible games is available on Atari's website). What's more, the 10 games (Adventure, Combat, Dodge 'Em, Haunted House, Maze Craze, Missile Command, RealSports Volleyball, Surround, Video Pinball, Yars' Revenge) are selected via four jumpers on the cartridge side, just like in the 1980s.

Atari presents new cartridges on its website, including a set of 4 sports games with CX30 manipulators (those funny joysticks with a wheel, the so-called paddle). However, the prices are not very appealing (one game will set you back around $70), so the buyer really has to be an Atari fan.

The console costs a little over $100, so you have to think carefully before buying. At the same time, I have to admit that it looks pretty good in my retro display cabinet next to the ZX Spectrum and the Atari 65XE. ■ Razor

On another subject

This insanely great Macintosh

author and translation: Leon

EDITORIAL

The Macintosh turned 40 this year. The first model was officially unveiled on 24 January 1984 and quickly became synonymous with a user-friendly computer and one of the symbols of pop culture and the entire decade of the 1980s.

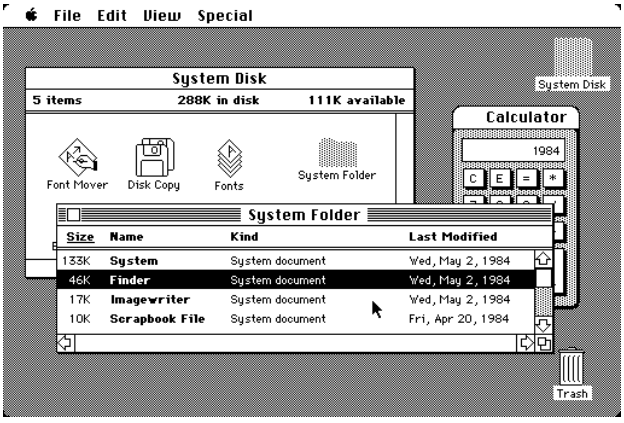
The first Mac was unveiled at Apple's annual shareholders meeting, held at the Flint Center at De Anza Community College. It was a real show with a spectacular and dramatic setting. The key moment was the unveiling of the latest product, done by none other than Steve Jobs himself, the charismatic yet controversial visionary and co-founder of Apple Inc. The event was preceded by an extensive marketing campaign, with Ridley Scott directing 'Apple Mac: 1984', a memorable Macintosh advertisement that was aired two days before the computer's launch during the Super Bowl XVIII, the NFL's annual championship game. The protagonist of this TV ad, one of marketing's milestones, was a young and energetic female rebel. With her hammer, she smashed the screen of an Orwellian Big Brother who was hypnotising a mindless crowd with his ideological message.

Mac Plus, the third model in the Macintosh family, launched in January 1986

Ironically, all this marketing hype worked out well for the Big Brother, an embodiment of IBM. Advertising costs increased the price of the Macintosh by $500 to $2495 (about $7317 today), making it a moderately attractive product despite the enthusiasm generated by its launch. As a result, the Mac was unable to threaten the market position that IBM had gained with the PC introduced in 1981. The high cost, for which Apple is still famous today, may also have contributed to the popularity of the next competing computer families, the Atari ST and the Amiga, which appeared in 1985. Based on the same processor as the Macintosh, but with greater graphics and expansion capabilities, and available at much lower prices, they quickly gained significant market shares.

On the technical side, the first Macintosh could be considered a pity by modern computer users. It had a mere 128KB of RAM and its CPU was the 7.8MHz Motorola MC68000 microprocessor. The Mac initially had no hard drive and its storage medium was 3.5" floppy disks of 400KB each. Graphics were displayed on a 9□ monochrome screen with a resolution of 512×342. The monitor was housed in a case with the central unit, to which a keyboard and mouse, and optionally other peripherals, were connected.

But if you look at the realities of the first half of the 1980s, the computer was an undeniable breakthrough. This was mainly due to its usability and aesthetics, which were meticulously overseen by Steve Jobs. The computer's appearance was close to the human face, and at first glance it evoked sympathy. But it was the operating system, known as Mac OS since 1996, that defined its character. It had a neat-looking graphical user interface (GUI) with elements that mimicked what we would normally find on a desktop. Pictograms, which over time became known as icons, represented sheets of paper, folders, floppy disks and other items familiar from office work, such as a calculator or a trash can. A major contributor to the graphical layer of the system was Susan Kare, who pioneered GUI design and a form of digital art known as pixel art. This talented artist created many icons and fonts, such as Chicago, Geneva, New York and San Francisco. Kare's icons were simple, understandable, somewhat playful and designed for a user who was not necessarily computer literate. Despite some variations in size, the icons all fit within the 32×32 pixel size. This provided a consistency of aesthetic that was lacking in the later Atar's TOS or Amiga's Workbench, which despite the use of colour could not beat the Macintosh system in terms of elegance.



Macintosh System Software 1.1

> *The computer was an undeniable breakthrough. This was mainly due to its usability and aesthetics, which were meticulously overseen by Steve Jobs. The computer's appearance was close to the human face, and at first glance it evoked sympathy. But it was the operating system, known as Mac OS since 1996, that defined its character. It had a neat-looking graphical user interface (GUI) with elements that mimicked what we would normally find on a desktop.*

The Mac OS was intuitive to use with a one-button mouse. Most importantly, it reversed the logic of working with the operating system. Whereas in command line systems you first type a command to indicate an action (e.g., cd, ls, mkdir for the familiar Unix system), the Macintosh system allowed you to select an object and then manipulate it to achieve a particular result. This natural, human way of using tools certainly contributed to the success of the Macintosh.

However, the story behind the creation of the Mac was not without controversy, as Walter Isaacson vividly recounts in his 2011 biography of Steve Jobs, which is highly recommended here. Suffice it to say that when Jobs took over the Macintosh project, he got rid of Jef Raskin, the originator of this computer and the head of the team working on it. This was Jobs' way of making up for his earlier failure when he was forced out of Apple's Lisa team. In the end, the Macintosh was a far cry from Raskin's original vision. The latter was realised during his time at Canon. However, the Canon Cat he developed there was not a market success. This corporate infighting pales in comparison to the circumstances under which Apple learned of the GUI's existence.

Mac LC III, released in February 1993

The knowledge came from Xerox, although to call it "the biggest theft in the history of the computer industry" is a bit of a misnomer. In fact, Steve Jobs was dazzled by the sight of a prototype operating system with a GUI during a presentation of the technologies being developed at Xerox PARC, the company's research and development centre. The meeting was part of an agreement between the two companies. Jobs agreed that Xerox would invest in Apple in return for showing him what was being done at PARC. To make matters worse, there was a group of people at the centre who had been passively sabotaging the demonstrations for Jobs and his entourage. But even they capitulated after a telephone complaint from Jobs to Xerox's headquarters in Connecticut. Interestingly, Xerox was the first to launch a GUI-enabled computer, the Xerox Star, in 1981. This machine did not make much of a splash, although it did pave the way for the subsequent triumph of computers with graphical operating systems. As you can see, having an original idea is one thing, turning it into a market success is quite another.

The stylish simplicity of Mac Plus peripherals

It should also be added that the Macintosh system was more sophisticated and, above all, more affordable than the Xerox Star, although not enough to dethrone the IBM PC-compatible range of computers. This also meant that it took many years for the Mac to become popular around the world, especially in less developed economies. In Poland, Macs could be found in the early 1990s in companies offering printing or DTP services. Macs also found their way into educational establishments. Among them was the secondary school attended by the author of these words. The school's computer lab was organised around several Mac LCs and a single Quadra, which deserves its own entry. Thanks to their presence in places like this, the Macintoshes have introduced many people to computers, which no longer seem like complicated machines operated by specialists. They also demonstrated how efficient and easy computer networking could be, thanks in part to the AppleTalk protocols and the AppleShare service. All this is part of the contribution of the famous Cupertino company, whose symbol is the bitten apple, to the popularisation of computer technology. ■

EDITORIAL

TALKING HEADS

A view into racing

Why do we enjoy racing cars in video games so much? Hmm... probably because we can push our gas guzzlers to the limit, burn rubber to the max, crush our opponents, or push them into a ditch. We love to live life at full throttle! The question is, do we like to do it from bird's eye view or from behind? Which of these racing genres suits us best? What are our favourite Commodore and Amiga titles? Ladies and gentlemen, start your engines and pedal to the metal!

Komek: First of all I have to say that I am more entertained by overhead racing. My favourite title is Carnage by Zeppelin Games. The game was released in 1993, at the end of the life of our beloved C64. Carnage is a murderous competition between four cars racing on small and winding tracks. Each victorious stage is rewarded with a large sum of money, which you can spend on upgrades and speed-ups for your car, or on mines to weaken your rivals. Race to the limit! There is no room for error in the race - a poor finish equals the end of the game.

The game has minimalist but cool graphics and good sound effects. Most importantly, it is engaging and addictive. Several of my friends can attest to this, and when they come to visit me, they want me to fire up Carnage! PS I'm attaching a photo from years ago of my friend Lukasz wreaking havoc in Carnage.

Jackal: Since firing up Lotus II on my Amiga, no other racing game has given me the same thrill as battling with nineteen other drivers, especially when you can split screen with another player. I'm so addicted to this game that I was recently browsing the used car ads and, in addition to a Mustang, I was looking for an Esprit.

▼ *Playing Carnage at Komek's place*

The view from behind the car is my favourite way to conquer racing games. I drive badly with the inside view. It takes all the fun out of it; it's like I'm sitting in a metal box instead of a sports car.

The other title that makes my skin tingle is Jaguar XJ220. Incredible driving atmosphere, on and off the road, spiced up with economic elements (car repairs, configuration changes) and, for me, a very high level of difficulty. In addition, a brilliant soundtrack (I recommend Moody Breeze or Troubled Journey) to top it all off.

There is one exception to the view from behind the vehicle that I love - the overhead view of the Micro Machines game. I have the impression that the title is reminiscent of Worms in terms of humour and atmosphere. Anyway, I had a great time playing it.

Don Rafito: It shouldn't come as a surprise, in fact I've said more than once, whether in my articles or in inter-editorial conversations, that I'm not a fan of bird's eye view games, but I can appreciate them if they're well made. Autotest Simulator (Daisysoft, 1990) on the C64 is such a title. Although it's not a typical racing game (you don't compete on a track against other racers, but with your own driving skills on a manoeuvring area), I spent many hours with it and still return to it from time to time. As for my favourite category, the bird's eye view racing games, I have very fond memories of Night Racer (Mastertronic, 1988). It was the first racing game of its kind that I ever played on a soapbox. I remember the slack-jawed amazement. On the Commodore, I was also impressed by Chase HQ (Probe Software, 1990), which I played on the Pegasus, and Lotus Esprit Turbo Challenge (Magnetic Fields, 1990), which ultimately shaped my taste in car games genre.

Before the Amiga came along, I was exposed to the PC 486 at school, where I was captivated by Lotus III: The Ultimate Challenge (Magnetic Fields, 1992). Once I fired up this title on my Amiga for the first time, which incidentally was a much better release than the PC one, it became my 'numero uno'. Lotus Turbo Challenge 2 (Magnetic Fields, 1991), although non-chronologically, also fell into my hands, but did not impress me as much as the third

▼ *Don Rafito trying out his homemade steering wheel back in the days*

part. Other Amiga titles that stick in my memory and that I like to return to are Xtreme Racing (Silltunna Software LTD, 1995), which showcases the capabilities of AGA chip-based Amiga, Flyin' High (PURE Design, 1997), which proves that it is possible to write car games on Amiga that are technically similar to those released on PS one, and Virtual GP (The Alien, 1999), the most well-made F1 simulator on Amiga.

PS Once upon a time I concocted an Amiga steering wheel out of a broken joystick and a small child's toy (as proof I attach a photograph taken with one of my first phones). What a nice experience it was to squeeze its rim while racing in Lotus....

Erik: Back in the days I used the play Pitstop 2 a lot. I liked it so much because you could play it with a friend using its split-screen mode and attempt to burst each other's tires by colliding. Bursting your opponent's tire would mean instant victory! Of course with great risk for your own tires, but that was half the fun. Another racing game I enjoyed was Kickstarter 2, a two player horizontal racer, where you had to drive across an obstacle course using your motor. Some obstacles could only be passed full throttle and others very slowly. Also there were jumps and you could do wheelies. But the best was when your opponent made a mistake and launched himself off his bike and into the mud. Yes, a little schadenfreude makes a good racing game a great one.

Sleeva: I have mixed feelings about top-down racing games. Maybe it's because I associate them with playing with toy cars on the carpet, but I think it's more a dissatisfaction with the confusing controls. I've come across games where the left-right movement of the joystick mirrors the movement of the steering wheel, rather than the movement of the car as a whole (you know, the car goes down, you move the joystick to the left and it goes the other way). Frustration rather than fun guaranteed. TPP, on the other hand, is a completely different story. Unsurprisingly, my favourites are Lotus (especially Lotus Turbo Challenge 2 and Lotus III: The Ultimate Challenge) and Jaguar XJ220, but also Cisco Heat, Top Gear 2 or Crazy Cars III, or rather Lamborghini American Challenge, released two years later, an improved version with an official Lambo licence and the possibility of two-player split-screen.

Beaver8bit: Back in the C64 days, I much preferred playing with an overhead view. When we have a bird's eye view of the track, we immediately know who is where and how much is there left to go. Also, we can see that the level authors had more creative freedom with the turns and loops. I would mention Jet Bike Simulator as a favourite from this genre. The game stunned with its speech synthesis, reading out its title just after launch.

As I got older, I began to appreciate the rear-view driving more and more, but this was mainly due to the higher quality graphics, where instead of a monotone sliding background and same-old corners, you could feel you were on the track.

I'm slightly surprised that no one mentions the view from inside the vehicle, as in Test Drive. Correction: it doesn't surprise me. I, for one, frankly hate such a view and if there is no option to choose a camera outside the car, then such racing game goes straight to the bin.

Tect: My favourite racing game is International Truck Racing. It's an unusual title for the C64 from Zeppelin games. It was developed in the same year as their other excellent game Carnage, which Komek mentioned. We sit behind the wheel of a red semi-trailer truck to compete against three other racers on tracks in different countries around the world. During the race, we must consider fuel consumption and damage to the vehicle as we push between trucks. If necessary, we must pull over for a technical stop. After winning a race, we receive money, with which we equip our vehicle with a better engine or afterburners. Despite the sparse graphics and lack of music while driving, the title was a lot of fun for me as a kid.

> The view from behind the car is my favourite way to conquer racing games. I drive badly with the inside view. It takes all the fun out of it; it's like I'm sitting in a metal box instead of a sports car.
>
> – Jackal

The next Commodore game I loved was 1985's Rally Speedway. We raced on suburban tracks against a race car controlled either by the computer or a second player. The title was notable for its dynamic gameplay and great driving model. Taking corners with the characteristic squeal of the tyres was very satisfying. We could also create our own tracks using the editor. I really enjoyed this option. Once I had created a track, I would save it on tape to compete with my friends in the neighbourhood. Since I built the track, I must know every turn by heart - this is how my friends argued their failures. Unfortunately, I couldn't persuade anyone to build their own tracks. The tracks I mentioned are from a bird's eye view. I can't help but admit that in my favourite games I prefer the overhead view.

Agnieszka FPWG: I would like to recall a true legend of racing games. There are many titles that have captured the hearts of gamers. One of them is the unforgettable Sega Rally, released by Sega in 1995. The success of the game was based on a simple but effective approach. The game offered exciting racing that was accessible to both novice and experienced gamers. Players could choose from a variety of tracks and cars with different characteristics. This classic rallycross-based racing game played a significant role in the development of the genre, and its influence can still be seen in contemporary productions. Any arcade veteran will tell you that Sega Rally Championship was a great game in the mid-1990s, competing with the likes of Ridge Racer and later becoming one of the few reasons to buy a Sega Saturn. A remaster of the game, titled Sega Rally Championship, was released in 2019, reminding the world why the series is legendary. Despite the passing of the years, Sega Rally continues to attract new players who want to experience the motor racing legend. Sega Rally sits on a shelf with other classics like Gran Turismo and Need for Speed. It may not match them in terms of technology, but it still holds the top spot as one of the most outstanding racing games of all time. New generations of gamers can still enjoy racing in Sega Rally, which is the best confirmation of the game's enduring value. Sega Rally is sure to provide an unforgettable experience and countless hours of entertainment, whether you are nostalgic or looking for a great game.

I had (and still have) a great time playing Slicks on the C64 with the overhead view and Stunt Car Racer with the cab view. In Pitstop II, I sometimes compete with myself to see if I can remember how to change tyres and fill up the tank as quickly as possible.
– phowiec

Retrobajtel: From my Amiga days, I remember Road Rash. Why is that? Because you could get on a motorbike, glide through the Californian wilderness and throw sticks, clubs, chains and whatever else was at hand at your opponents. If necessary, a hard kick and the guy's gone. Although I wouldn't hurt a fly in the real world, I had a great time playing this game. Bad boys, illegal racing and the adrenaline... I raced, won, often dishonestly, and invested the money I earned in new bikes. That was my gaming dark side... Another game I remember was Lotus. Not that I was a big motoring fan and loved racing or anything, but this game was great for two reasons: it was split screen, so I could race with a mate, and there was... a radio. The radio is what I remember most. How cool it was that I could change tracks... Not just any tracks. Amiga is still a musical delight to this day. If you woke me up in the middle of the night and said "Lotus", I would answer "radio".

I would almost forget to mention my all-time favourite racing game - Stunt Car Racer. Racing on a track tens of metres above the ground with no barriers. Hours of skilful grinding to keep from flying off the track at ever-increasing speeds; exhaust fires, jumps over precipices and hard landings either on or off the track, several dozen metres below; and those distinctive chains by which a special crane lifted us up after a crash and put us back on a ramp. My memory is filled with many of these games. But there is one title I cannot recall. It was a time trial, quite difficult because I could only reach the second level. First there was a car ride that took you to a pier where you changed to a jet ski and continued. That was one of my first Amiga games.

As far as racing games go, I've always been a fan of first-person view simulators. And preferably if you could turn into a side street somewhere. And there was Test Drive... and I liked it. But unfortunately, you couldn't go anywhere off the track. And then there was Power Drive, which was a top-down game where you could race in an iconic Cinquecento. For this game I needed a modified Amiga with a switch to allocate extra memory (CHIP, FAST/SLOW). My dad did the conversion, and I was able to race the "Cinquecento". Those were the days.

Phowiec: From behind or from a bird's eye view? It doesn't really matter. I had (and still have) a great time playing Slicks on the C64 with the overhead view and Stunt Car Racer with the cab view. In Pitstop II, I sometimes compete with myself to see if I can remember how to change tyres and fill up the tank as quickly as possible. That's where the rear-view camera comes into play. The important thing is that it is fun for me. My fascination with car games hasn't waned over the years, even though my knowledge of cars doesn't follow suit. If it's about racing, that's enough. Death Rally, Carmageddeon, Absolute Drift, Forza Horizon - I've always found the magic ingredient in each of these games and I'll happily return to any of them, even if it's crap like The Last V8. Nostalgia keeps me attached to this title. Probably because it was my first game (and an original at that) on my C64.

Razor: My first attempts at racing were, of course, on the C64 and Pitstop II by Epyx. What a thrill it was when we took turns with our friends trying to get the best lap times (we even had a notebook to keep track of). Then there was something better - Stunt Car Racer with pseudo 3D. Great fun on breakneck tracks, driving upside down, barrels... And of course the absolute hit, Street Rod, street racing straight from 1969, beautiful American cars that could be upgraded, very good graphics and a view from behind the wheel. Then there was racing on PC and consoles, until all kinds of Need for Speed came out. Agnieszka mentioned Sega Rally - that was indeed something, but on consoles I liked Sonic & Sega All-Stars Racing (now available on older generations of Xbox and PS) - a simple game that gave maximum fun racing against characters from other Sega games. Despite the passing of the years, I still come back to racing games that can give you out of the box fun, without any additional fiddling with the settings.

Maciek: Like Razor, my absolute first contact with racing and the C64 was Pitstop II. I consider it one of the better games of its kind for that computer. Its minimalism goes hand in hand with the sophistication characteristic of the Epyx. The split screen and the ability to play with your brother at a time when there was no network play (because there was no network to speak of) put it on

a par with such classics as Barbarian and IK+. It turns out that there was a second top view game with a similar competitive mode: Commodore's Rally Speedway. The competitive mode also required a degree of cooperation - you had to stay close enough to your opponent to avoid falling off the screen. So the natural hardware limitation (no split screen) created a unique form of entertainment.

Tomxx: I started with Buggy Boy. Returning to the C64 years later, I find that the game has aged well. The other title that impressed me at that time was of course Accolade's Test Drive. I had a floppy disk drive. So from a young age I had the opportunity to play its second instalment - The Duel: Test Drive II. I still remember the atmosphere of driving past cliffs and running from the police. This game has aged less subtly, though, and today it's a bit boring in its slowness.

To get back to your question, I like overhead view games, and for me the pinnacle of the genre was the first edition of Grand Theft Auto. The gripping and dynamic nature of this 1997 PC game has long remained in my memory. From our backyard, I like to return to Championship Jet Ski Simulator, a game I briefly recall in a column in the current issue of the magazine.

Monka: I like both points of view, and whether a game is good or not depends entirely on the concept and execution. Looking back, I think I was most impressed by the ability to create a sense of momentum and speed. Despite appearances, this was not easy to achieve given the technical limitations. The first rear view racing game I played on my C64, Pitstop II, was later only surpassed by the phenomenal Power Drift and Out Run Europa. These three games differed in virtually everything except the view and the feeling that you were actually racing in cars that reached the speeds seen on the screen.

With an overhead view, the only C64 games that gave me the same feeling were the forgotten and underrated Warm Up and the fantastic Slicks.

Leon: For me, the enjoyment of the game is more important than the form of the world presented in a given title, although I have certain preferences in that regard as well. I have usually been fascinated by techniques that create a sense of spatiality. This kind of effort seems to me to be more ambitious than the simple approach of flat representation. Hence, ever since I saw and played Pole Position in an arcade in the 1980s, I've definitely grown fonder of the behind-view style. This belief was reinforced in the next decade by the Amiga's big hits - games from the Lotus series and its rival Jaguar XJ220. However, these popular pseudo-three-dimensional racing games were about to meet a worthy opponent. We're talking about Super Skidmarks, with its gorgeous, albeit different, isometric graphics, incredible dynamics and sensational playability. On the A600, this title even outperformed the aforementioned and slightly older rivals in terms of popularity. Without a doubt, Super Skidmarks is a game that provided a lot of fun even in the second half of the 90's, which was rather difficult for the Amiga. I do not rule out the possibility that during my various discoveries in the world of retro-computing, I will come across a game from years ago with a bird's eye view, as one of the previous speakers put it, which will be a source of many positive emotions.

▼ *Buggy Boy (Elite, 1987) and The Duel: Test Drive II (Accolade, 1989) – Two completely different racing games on the C64, which also aged very differently*

Ari: For me, I've never had a favourite view in racing games. I think it is not the most important one. There are many factors that influence the enjoyment of a game and they just add up to whether we like the game and whether it's fun to play. On the C64, my best memory is of Power Drift and the excitement of the tracks going up, down and sideways. I also had a great time playing OutRun, Pitstop II and Slicks. As for the Amiga, it's clear - mainly the Lotus series, but Crazy Cars 3 was also good. I also have a lot of fond memories from the PC DOS days. For a long time Stunts was the undisputed leader, then my dad and I switched to cars in the first NASCAR. I returned to racing with Colin 2.0 and what I consider to be the best racing game ever, Richard Burns Rally. The racing game I play most at the moment is Mario Kart on Nintendo Wii, because my daughter really likes it. We get behind the wheel in front of the screen on average once a month and it still gives us as much pleasure as it used to.

As you can see, I've played a lot of racing games with a different view. But I think the most important thing is still the joy of completing another lap, setting another record, or making it to the finish line in one piece.

Louie: It is overhead racing for me. It provides a better sense of speed, control and there is more of the game to see. Like many of the others Pitstop II is a stand out in this genre, not only because of the split screen 2 player perspective but also because the tyre damage mechanics, pitstop feature and multiple tracks. Buggy Boy would also be a stand out as it contains the perfect arcade blend of racing, collection and jumps making it such great fun to play even to this day. Such as shame that modern day coders tend to avoid the genre. ■

JOHNNY PRESENTS

Bobr Album

At the end of 2023. Bobr Games released an album of covers and artwork related to its products. In addition to reproductions of images from game boxes, there are also sketches and unused designs. The publication also contains texts with, for example, interesting facts about the covers I drew.

For today's issue of K&A plus, I decided to prepare a board that gives a behind-the-scenes look at the creation of an album cover. Here we have a finished image with a beaver, which might be a little associated with anthropomorphic characters from Disney cartoons.

In this compilation we see a preliminary sketch expressing an idea, but also a drawing in ink. A keen eye will notice that the desk was to include a joystick, which we replaced with a single button.

In addition, a graphic from Bartosz Żołyński's game 'CommodoRex' appeared on the screen. Thanks to Łukasz Bobrecki for involving me in the creation of this album and for his support in the creation of today's edition.