FIRST ISSUE OF 2022 - HAPPY NEW YEAR

RetroMagazine World

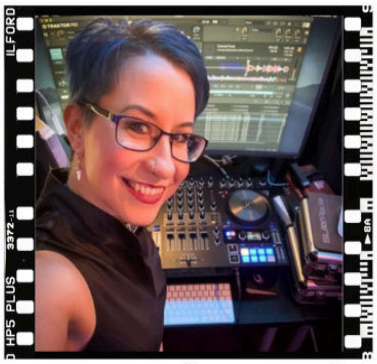
future days are back

ISSUE 12 JANUARY 2022 - WWW.RETROMAGAZINE.NET - FREE PUBLISHING



Hardware review MAGNAVOX ODYSSEY

Sonic runs on COMMODORE 64!



Interview with ERIKA WITTMANN





ON SEGA DREAMCAST

Programming GEOS in BeckerBASIC - RetroN sq

C128 characters in 80 colums... and much more

Lots of REVIEWS!!!

- What are they celebrating?
- Their planet made a full circle around their star...
- I told you they are not intelligent!
Common alien's joke about new year's eve

And this 2021 has also come to an end. Another year spent in the grip of COVID. In my last editorial in December 2020 issue, I had hoped for an end to the pandemic emergency.... Maybe I blew it, so I won't make the same mistake this year.

As I was saying, we are at the end of the year, in that time when we take stock of what has been done during the 12 months that will soon come to an end. In the past, I've listed the statistics, lavishing praise on those who collaborate with this editorial adventure on a permanent basis. This time, however, I won't dwell on the digits and I won't make budgets....

- So what? What will you tell us?
Well, I want to tell you about our satisfaction in having realized and in continuing to realize our dream of when we were kids. All the guys on the editorial staff of RetroMagazine World, dreamed of being part of a magazine like the ones they used to buy in the 80s and 90s. At this point I think we can say, without fear of contradiction, that we have carved out our own space in the world of retrocomputing and digital publishing.

- Oh-oh, here comes the 'jingle' of how good and beautiful they are...
No, not at all, instead I want to emphasize how important it is to work hard together to make something we care about. I want to praise not so much the result - you dear readers will take care of that - but the commitment that each member of the editorial staff puts into creating something unique in the national and international panorama. The time that each of us spends to study, try new things, test hardware and software, write an article, make sure it is as error-free as possible ... It's really amazing how, after 5 years, we are still here, with the same will and the same impulse of the first issues. Of course, some have slowed down their production and in the meantime new forces have taken their place, but the core of RMW is still strong, active, pulsating and able to surprise me and, I hope, you too with every issue.

All this effort, certainly deserves to be acknowledged and I hope you will join me in thanking all the staff of RetroMagazine World, which I am honored to represent in this editorial.

Before I leave you to read this issue 12-EN, let me wish you all **Happy Holidays** and a wonderful **New Year**!

Francesco Fiorentini

TABLE OF CONTENTS

◊ Famicom Disk System - the (un)protections Pag. 3
◊ RetroN Sq Pag. 6
◊ Magnavox Odyssey Pag. 8
◊ Optimizing the calculus of PI with MonteCarlo Pag. 14
◊ Programming in GEOS with BeckerBASIC Pag. 18
◊ Redefined characters for 80-columns C128 Pag. 28
◊ A Christmas Fairytale Pag. 31
◊ BASIC Fractals in 8-bit sauce - Part 2 Pag. 34
◊ Commodore 64's guidebooks re-edition Pag. 37
◊ Welcome to the Palace of Magic Pag. 38
◊ Japan 18th issue: Nintendo VS all Pag. 41
◊ Retro Interview: Erika Wittmann Pag. 46
◊ Armalyte for Commodore 64 Pag. 49
◊ Galaxian, from 1979 to 2021 Pag. 52
◊ ListAmiga - the most overrated 5 Amiga titles Pag. 54
◊ Blade of Agony (PC) Pag. 59
◊ Sonic the Hedgehog (C64+REU) Pag. 60
◊ Slow Mole (NES) Pag. 62
◊ Snow Bros (GameBoy) Pag. 63
◊ Gaiares (MegaDrive) Pag. 64
◊ Metal Slug 6 (Dreamcast) Pag. 66
◊ Tecmo SuperBowl (NES) Pag. 68
◊ Jazz Jack Rabbit (GameBoy Advance) Pag. 70
◊ Hallowed Knight (ZX Spectrum 48k) Pag. 71
◊ Over Horizon (NES) Pag. 72
◊ Little Medusa (Super Nintendo) Pag. 74
◊ Xenocrisis (MegaDrive) Pag. 76
◊ Bishoujo Senshi Sailor Moon R (GameBoy) Pag. 78
◊ Vastar (Z80) Pag. 79
◊ Strider (C64) Pag. 80

People involved in the preparation of this issue of RetroMagazine World (in no particular order):

- Alberto Apostolo
- Dr. Andrea Q.
- Carlo N. Del Mar Pirazzini
- Daniele Brahimi
- Mic the Biker Novarina
- Francesco Fiorentini
- David La Monaca
- Leonardo Miliani
- Giuseppe Rinella
- Barbara “Morgana” Murgida
- Ermanno Betori
- Takahiro Yoshioka
- Roberto Del Mar Pirazzini
- Ingrid Poggiali
- Gianluca Girelli
- Michele Ugolini
- Maurizio Diamanti
- Robin Jubber
- Carlo Luciano Bianco
- Alessandro Gatti
- Cover image: **Giuseppe Mangini**
- Cover layout: **Carlo N. Del Mar Pirazzini**

Famicom Disk System - the (un)protections

by Dr. Andrea Q. - www.retrofixer.it

Youtube channel: https://www.youtube.com/channel/UCEw0CQ8LKyA9jVvWXkEwp4Q

In the previous article we talked about the protections inserted by Nintendo in its "export" NES console; today we will see how the FDS add-on device behaves.

The Famicom Disk System (product code FMC-001) was released by Nintendo in February 1986 as an add-on for the Famicom (the Japanese version of the NES for those who do not remember) and allowed the reading of games, as well as cartridge, even from floppy disks. In the image below you can see it lying below the Famicom with its expansion cartridge (black) inserted:



Below in the "integrated" (Famicom+FDS) Twin Famicom version produced by Sharp:



This peripheral actually had a precursor, the Nintendo Data Recorder, created in 1984 for the Japanese market, but it was unprotected:



It was used to store programs written for the BASIC application or to save levels created for a handful of titles that supported it (Castle Excellent, Excitebike, Mach Rider and Wrecking Crew).

The purpose of the FDS instead, in addition to the possibility of saving the progress made (one of the very first titles released was in fact The Legend of Zelda!), was to increase the space to produce and distribute games, space that at the time was still limited by the small ROM cartridges! Together with the drive was supplied an expansion cartridge containing RAM chips and an ASIC chip with DRAM controller, IRQ hardware, a hardware for sound generation, a serial interface for the drive and a parallel port.

To officially obtain the games loaded on diskette you had to go to the "kiosk" that had a style similar to the vending machines of today.



If you already had a disk (empty or already written) this could be reused for 500 yen, if you also wanted an empty disk you had to add another 2000 yen.

FDS PROTECTIONS: FORMAT

The hardware of the machine has a floppy drive 2.8" x 3"

and 64Kb of storage space per side, so 128Kb total; the storage format on disk is that of QuickDisks (the same format as MSX) but using a plastic case 3 "x4" instead of the standard 3 "x3". Not being something "standard" the copy was already made complicated: in fact it was necessary to find a system to use more "common" disks like the "normal" QuickDisks.

In addition, the writing "NINTENDO", on the back of the floppy, was hollow and the reader did not accept disks that did not have this "engraving".



But the hackers did not stop at that, starting with a standard QuickDisk that you can see next door.



Systems were produced to adapt this disk to the FDS format, which consisted of cutting away the "fins" of the standard disks, applying a rear "extension" and attaching a piece to "simulate" both the correct length and the presence of the engraving of the NINTENDO logo (in particular, the physical control was on the letters I and N of the writing):



SOFTWARE CONTROLS

There are several controls performed by the console:

- the system looks for the 14 bytes string "*NINTENDO-HVC*" at offset 0x01 of the disk and if it doesn't find it it returns an error;

- 224 bytes at address $2800-$28DF of the PPU must coincide with those of the BIOS stored at address $ED37;
- the first file of a disk is called KYODAKU- (meaning "approved") and must be present otherwise an error is returned;

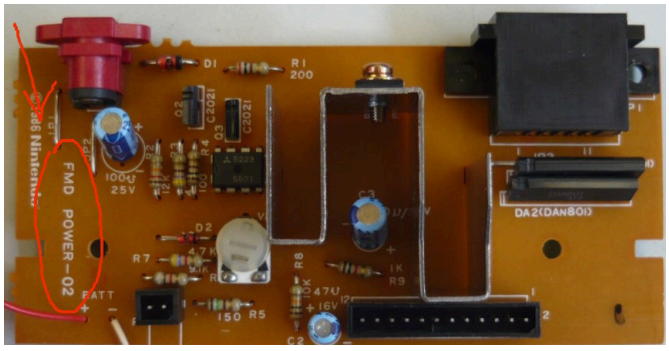
- some developers could implement checks on hidden files that the bios did not read, which were still "searched" by the game and if not found would return an error.

These checks could have been defeated by obtaining a 1:1 copy of the original floppy but ... how to do since some drive controllers, by factory, have been built adding a dedicated logic to not allow this function? The answer came in a short time: using a hardmod on the FDS that allows the complete read/write bypassing the controller of the board (it's called "FDS write mod") and using a dedicated software to make the dump.

To summarize: there are basically 2 versions of the controller, the FDP3206P and the FD7201P; in the 3206P, if the command "write a file" is sent, the system works but if the command "write the whole disk" is sent (with which all sectors and any hidden files can be copied) the system disables the writing of the head that continues to "not write" in silence: when the write check takes place the system obviously reports an error because nothing has been written ("Error 26: Could not write to disk card"). The modification allows you to bypass this factory-imposed "limit".

HARDWARE CONTROLS

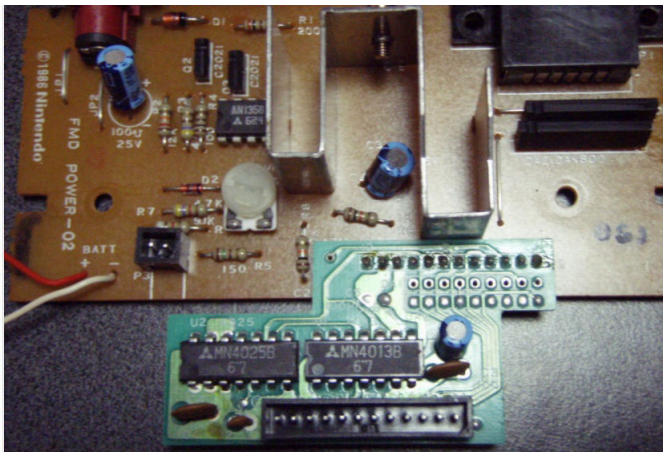
Later Nintendo introduced a further level of control that still excludes writing (even if the controller is an FD720P !) from some hardware revisions of the power board; this system component can have 5 possible revisions, each identifiable by a specific string written on the PCB that you can see circled in red in the picture:



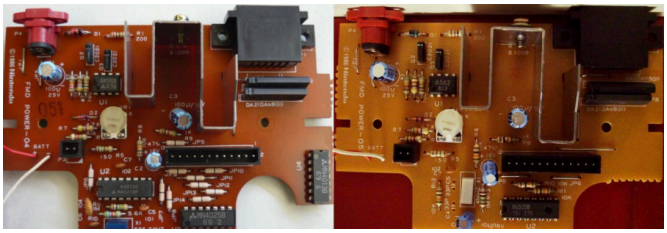
FMD-PWER-0X where instead of the X there can be a number between 1 and 5; revision 1 is unprotected, revision 2 may have it sometimes not, revision 3 is probably unprotected while revision 4 and 5 definitely implement protection.

Revision 02 with protection is easily identifiable because

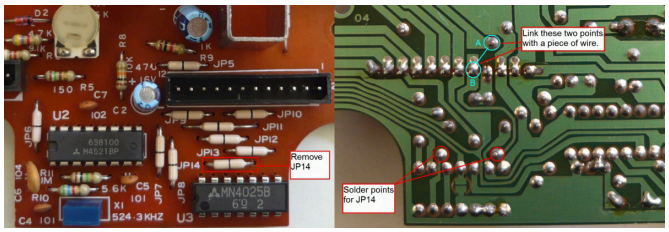
it has an additional board that must be removed (and the connector must be re-soldered) to make it look like an unprotected rev.02:



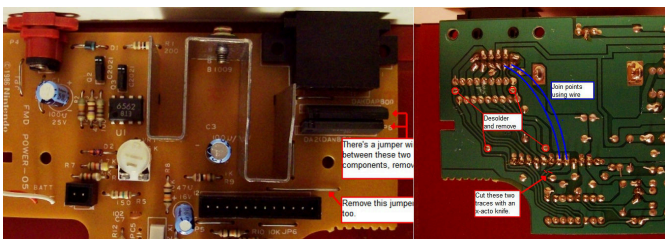
Rev.04 and 05 instead have the additional part integrated in the PCB:



thus making some track cutting, removal of some components and jumpering necessary to get it right.
Rev.04 - remove/disconnect component JP14: this totally disables the protection; however a jumper must be created between points A and B of the board to restore the write function.



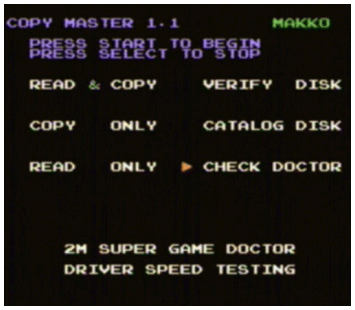
Rev.05 - remove/disconnect the jumpers highlighted in the picture on the left (the top one is hidden between the 2 black plastic components) - the corresponding points on the other side of the PCB are highlighted in red with the label "Desolder and remove":



then cut the 2 small tracks indicated by the 2 small "red dashes" in the image on the right and finally connect 2 wires where indicated by the blue lines to restore the write function.

There is also a software (Copy Master NTSC version) that works without having to make these mods but is not able to produce a perfect copy because it is not able to write *NINTENDO-HVC* or any hidden files so, not copying perfectly 1:1, could produce copies not working.

There are also dedicated hardwares that work by using 2 FDS connected together as the Dubbing Boy II, or the Famicom Disk Backup Unit, both with its own software for copying (like the one on the side) with which you can use one FDS to read and one to write data on a destination disk.



So you can see how the evergreen cat-and-mouse chase between physical protection mechanisms (proprietary hardware with specific anatomies) and digital protection mechanisms (controls) VS the corresponding systems to overcome them is starting to take shape!

So let's see how we are laying the foundations for that magical world that has held and is still holding sway among modification enthusiasts around the world! Shall we call them hackers ? Shall we call them devs? Whatever name you choose one thing is sure: as long as there will be a protection there will always be someone with the desire to try to overcome it!

In the next article:
GAME BOY/COLOR/SUPER PROTECTIONS!

ATTENTION: LIABILITY DISCLAIMER

The information contained in this article is for informational purposes only. This documentation is not guaranteed to be error-free. If this information is used to modify your hardware, it is your responsibility to take all necessary emergency, backup, redundancy, and other measures to ensure safe use. RetroMagazine World assumes no responsibility for any damage caused by the use of the information in this article.

RetroN Sq - How to play all Game Boy titles on your TV screen!

by Takahiro Yoshioka and Carlo N. Del Mar Pirazzini

The RetroN Sq is a tiny cube-shaped console made by Hyperkin, a company known for producing retro consoles. The console can play Game Boy, Game Boy Color and Game Boy Advance games up to 720p on your TV.
A real dream for all those who own the old Gb cartridges. The machine was released on March 25 of this year at a price of about $ 75, obtaining a good sales success in the first week unfortunately arrested by some technical issues such as incorrect proportions, slow framerate and overall poor performance especially with Game Boy Advance games. Thanks to a series of firmware updates Hyperkin is able to solve the problems by making a patch that can correct the framerate and modify the loading of ROMs.

One of the best things about this little cube is its ease of use and small footprint. Just plug it into your TV, connect the power and pad and you're good to go. The sq is also about the size of three Game Boy Advance's put together, making it compact even for travel and doesn't take up much space. It comes with an HDMI cable, a USB-C cable, a 512 MB Micro SD card, and a fairly sturdy and convenient Hyperskin Scout usb controller. It mimics the Snes controller with good tactility in the main buttons and the two somewhat spongy but responsive back buttons.
But how does it work? Let's start by saying that it's not an FPGA spot.
Sq uses software emulation. Any game cartridge inserted into the console will be downloaded to the Micro SD and stored. So when you go to play the first time, Sq will take a while to download the game, later plays will be shorter to load (conveniently from a slider menu we can select stored titles).
During the dumping process, the copied save file will also be displayed.

Game Boy and Game Boy Color games run very well indeed at 720p in their native aspect ratios. The pixels are sharp and the colors really stand out. Through a switch on the back we'll be able to decide the game size of the screen.





Testing it with Mario Golf and Pokemon Blue we had no problems with frame drops, strange pixels or sound errors.

The original Game Boy games will be displayed with the same color palette as you would have by plugging them into a GBC. Some titles look strange on a large screen, as they are not designed to be viewed on a 50-inch TV, but everything is remedied by adjusting the size.



Speaking of the already mentioned Game Boy Advance compatibility, the latest patches have modified and improved a lot, but there are still some problems. If on the side of the frame drop everything is perfect, the audio side has some unpleasant noises and a strange effect of sound slowdown. We sincerely hope that in the next updates the situation will improve permanently.

The save files are another topic I want to talk about. When we insert the game into Sq for the first time, as we've already mentioned, it will be copied along with the save data from that cartridge to the Micro SD card. So, every time we save the game, Sq will overwrite the data on the SD card, but not on the cartridge. If you go to extract the cartridge to play it on your portable device or other hardware we won't see any progress in the save state. The guys at Hyperkin decided to specifically implement this system to preserve the original save data. I appreciate the sentiment, but not being able to be able to play your progress when you're on the go is a bad choice.

The Nintendo Switch is proof of just how important it can be to keep playing when you're away, especially when playing titles like Pokemon.

Our verdict

In summing up, Hyperkin has made a valiant effort to create a low-priced console that is easy to use and most importantly dedicated to games from the Game Boy world. Too bad for the save state problem and for the not perfect emulation of the GBA.

If you are interested in the product you can buy it from this site:
https://www.hyperkin.com/retron-sq-hd-gaming-console-for-game-boyr-game-boy-colorr-game-boy-advancer-hyper-beach-hyperkin.html

We thank Hyperkin for having provided the console for the test.

OUR FINAL SCORE





Magnavox Odyssey

by Leonardo Miliani

The Christmas holidays, as ever, are for all the kids the moment in which they dare to ask for that gift that they don't have the courage to ask for in other periods of the year. And, as ever, one of the most desired objects under the tree is a video game console. At Christmas time even parents who must provide the economic outlay necessary to meet the demands of their children hardly say no... Nowadays the purchase of a console as a Christmas present is almost a normal fact, like the purchase of a common household appliance... You simply go to a specialized store, you choose the model that intrigues you the most and you buy it...

But has it always been like that? Surely not, because there was a time when home consoles didn't exist. And, if there was a time "before the appearance of consoles" and a time "after their appearance", then surely there was also an "instant zero", a "big bang" from which everything was born. And do you know what is the origin of those little electronic contraptions that cheered our days as kids and that today cheer our children? Many will think it was the Atari VCS but this was a second generation console... There was someone else before it... Pong? No... except that it was born as an arcade before arriving in the homes of ordinary citizens as a machine that could be connected to the TV in the living room, but it's close... that someone else is the Magnavox Odyssey!

Origins

The Odyssey was the first game console designed and marketed for home use, sold to entertain people at home. In fact, it was the first console designed to be connected to your television set, and the first device that could be programmed using "cartridges" to offer multiple games in a single device. It also preceded the already mentioned Atari Pong by a few months, having been marketed since the end of August 1972, although we are talking about the arcade version because the domestic Pong will not arrive until 3 years later. But how was this console born? To find out we have to go back even further...

We are in the 60's and we are at Sanders Associates, Inc., an American technology development company that supplied the US Defense. Ralph Baer (fig. 2), a naturalized German-American electrical engineer whose family, of Jewish descent, fled Nazi Germany in 1938, works at the company. Baer, after having served in the American army, continued his studies and became an electronic engineer. He then worked for several companies until, in the '50s, he was hired by the aforementioned Sanders. Over time



Fig. 1: The Magnavox Odyssey with a controller (photo by Evan-Amos)



Fig. 2 - Ralph Baer (foto: Michael Schilling)

he improved his position until he became head of the department of development of military equipment: in 1966, under his direction, there were about 500 people including engineers and various personnel. At that time, his department was working on visual devices, and Baer began to develop the idea of using home TVs not only for viewing television signals but for other purposes as well. The idea of using some kind of device to play electronic games takes shape in his mind. The commercial potential is there: at that time there were 40 million TV sets in the United States. Baer presented the idea to his superiors but received a flat refusal: after all, Sanders developed military equipment for the Army, and could not waste time with such things. But Baer doesn't lose heart. His position allows him to move an engineer or two to a minor project without much explanation, and he involves engineers Bill Harrison and Bill Rusch in the development of his idea. The fruit of their labors matured on May 15, 1967, when the first game in the history of computing was played on a common TV in what can be considered the first home video game (previously there had already been other video games, such as "Spacewar!" in 1962, but until then had been run on large mainframes), the "Bucket Filling Game". Harrison and Rusch, following the guidelines of Baer, realize a device that can be connected to a common TV that divides the image of the screen into two horizontal halves: the upper half blue and the lower half black. To complete the picture, an adhesive film overlays the screen with the image of a scene of a house fire being extinguished by a fireman pumping water. Each of the 2 players presses a button: the first to pump water (the blue half that grows), the second to remove it (the black half that grows). After a few demonstrations, Sanders' top management approved the project and provided a small amount of funding to allow Baer to continue its development. The next step is the famous Brown Box, so called because of the color of the scotch used to wrap it and simulate the grain of the wood (fig. 3). This is a more refined device than the Bucket Filling Game, capable of generating and moving square shapes on the screen: the two larger ones are driven by the players through potentiometer commands while the smaller one represents the object they interact with. The device is made in such a way that it can be "programmed" from the outside: by means of switches one can choose different "games" as well as modify their dynamics, in order to offer games that can recall well-known sports or other entertainment. It is also developed what is to all intents and purposes the first "light gun" in the history of video games, i.e. an "optical gun" used as a tool capable

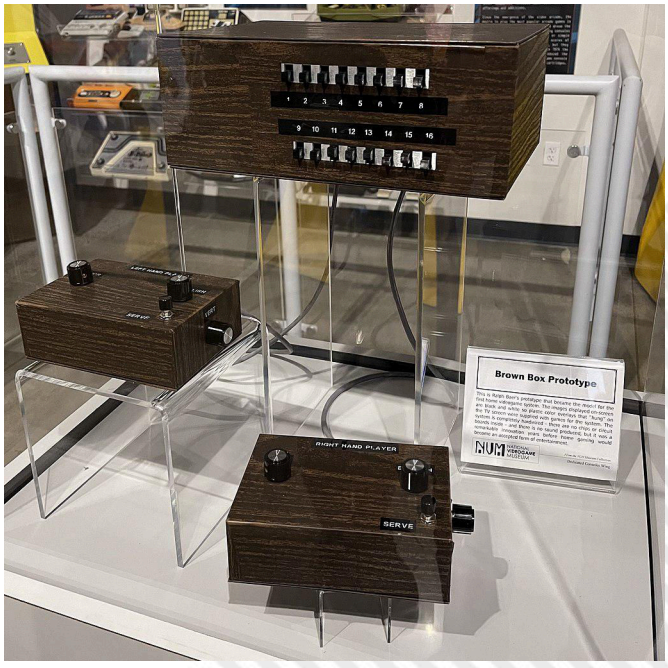


Fig. 3 - a prototype of the Brown Box, so called because of the brown scotch tape used to simulate the grain of the wood



Fig. 4 - Interior of the Odyssey. You can see the circuits made with discrete components (photo: Evan-Amons)

of "seeing" a target on the TV screen. In order to develop the idea, a real toy gun is purchased, modified by mounting an optical sensor on the tip of the barrel, with which the player must "shoot" at a moving target on the screen.

The Brown Box was completed in 1968 but, for two years, it remained only a prototype because, ironically, no company operating in the cable TV sector was interested in it. At the beginning of the '70s of the 20th century it was decided to show it to several TV manufacturers and, finally, the turning point came: Magnavox showed interest and signed an agreement to obtain licenses for the prototype in 1971. The console is officially presented in the spring of 1972. The Magnavox Odyssey was born.

Technical Features

To move from the Brown Box to the Odyssey, Magnavox creates a special development team that also works with Baer and Harrison. The circuitry is revised and the ability to generate color images is removed because at the time color TVs were still considered luxury items, and therefore color was considered a superfluous accessory. Moreover simplifying the video generation circuit by removing the color also simplifies the process of requesting the necessary authorizations from FCC (the U.S. Federal Communications Commission) for the marketing of equipment capable of generating radio and television signals. Despite the fact that in those years integrated circuits started to become popular, it was decided to maintain the original circuit based on discrete electronic components (such as resistors, capacitors and transistors) to reduce production costs (fig. 4). It was also decided to modify the game selection system of the console, eliminating the switches and creating external cards to be inserted in a special slot, to be sold separately. These are not real "game cartridges", as will be realized for future consoles, because they do not contain active electronics but only a series of tracks that modify the internal connections, changing the game mode as did the switches on the Brown Box (fig. 5). The console is equipped with a plastic container with an



Fig. 5 - One of the "game cards" (foto: Evan-Amos)



Fig. 6 - some of the films to be applied on the TV screen (photo: Evan-Amos)

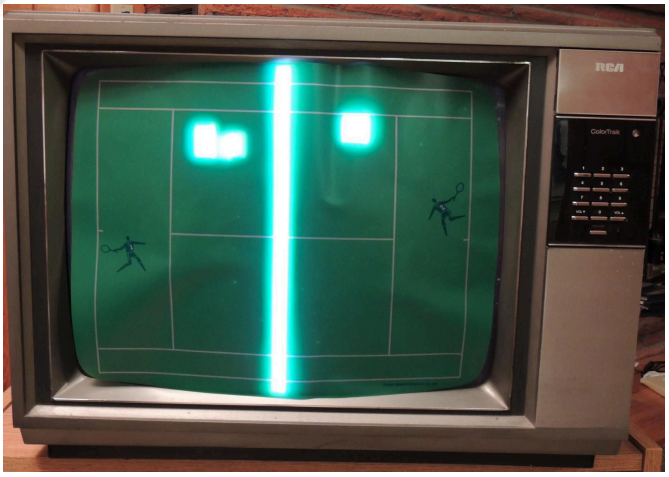


Fig. 7 - The game "Tennis" with its film. Note the static profiles of the two players at the sides and the lines of the court (photo: Mr. Glitch's Retro Reviews)

elongated shape and connected to 2 controllers connected by cable equipped with 2 potentiometers to move the corresponding square respectively on the vertical and horizontal axis, and a reset button. On the controllers there is a curious writing "English": it is not used to set the language of the console but to give an effect to the ball (an Anglo-Saxon term derived from billiards). The colors chosen for the console are white, black and brown (fig. 1).

The games themselves are revised, although they are based for the most part on those initially developed by Baer, Harrison and Rusch. The console is marketed with 13 "game cards" corresponding to as many games, including Table Tennis, Ski, Simon Says, Tennis, Analogic, Hockey, Football, Soccer and others. Other game cards are later marketed separately, while the optical rifle is sold in combination with a set of 4 game cards, all offered under the name of "Shooting Gallery". Altogether there were 28 different games released in the period in which it remained on the market.

If we think of the Magnavox Odyssey as an ordinary console we are making a big mistake. The Magnavox Odyssey is particular because its capabilities are very limited. The graphics themselves consist of simple lines and squares that appear on the screen with the graphic detail added by applying transparent films to the TV screen. These films also allow you to "create" a new game from another: for example, the game of ping pong, without film, is transformed into tennis by applying a green film with the lines of a classic tennis court and the silhouette of 2 players with rackets at either end (fig. 7). It is not even capable of keeping score so players are asked to mark their own on paper as the games progress. Same thing with the games included in the optical rifle package, which rely on the honesty of players because many find that pointing at a bright light source, such as a light bulb, tricks the optical sensor into marking the target as hit. In addition to not keeping score, the console is not even able to handle the event of the scored point: for example, if in ping pong the ball exits the screen, it is required that the player presses reset to restart the game, thus returning to the "hit".

Since the term "video game" is not yet in use, Magnavox presents the console as "the electronic game of the future" (in the original advertisements "the electronic game of the future"). (in the original advertisements "the electronic game of the future") and, in order to avoid that people do not clearly understand what Odyssey is, it equates it to something familiar, i.e. the classic board games, and distributes it with included in the package paper money, chips, dice and other accessories to be used in some of the games offered by the console. For example, in the game "Roulette" the player must spin a rudimentary ball that stops on the reproduction of a casino roulette applied on the screen with one of the inevitable films and manage the bets with the game board (fig. 8).



Fig. 8 - Paper money, dice, playing cards, chips, game cards and boards included in the Odyssey's packaging (photo: Evan Amos)



Fig. 9 - The Magnavox Odyssey 200 (photo: Elvis Untot)

After the presentation, the console went on sale at the end of August 1972 at a price of $99.95: an important figure that, today, would correspond to more than $600. In mid-September, Magnavox began an advertising campaign to publicize the console both in print and on TV channels. Magnavox sells the console only through its retailers, hoping that people who come into stores for the console will then be interested in some of its other products. Sales are good, even though people perceive the fact that the console can only be found in Magnavox stores as working only with TVs of that brand: to clear up this doubt, at the beginning of 1973 Magnavox itself is forced to publish ads explaining that the console works with any TV, be it black and white or color, of any brand. At the end of 1973 the console was lowered to 50 dollars and from 1974 exported to other countries, including Europe. The console remained on the market until 1975, eventually selling just over 350,000 units. The Shooting Gallery was less successful, selling only about 20,000 units.

Lawsuits

The console did not have many clones because technically and commercially not very interesting. Nevertheless, one of its games in particular, Table Tennis, inspired a person who based his fortunes on a replica of it. We are talking about Nolan Bushnell, founder of Atari, a company created in 1972 after he attended a presentation of the console and saw that game in action. Having already created with Ted Dabney what is considered the first arcade game in history, Computer Space, he saw the commercial potential of Table Tennis and decided to create a clone, which was marketed in late 1972 as "Pong". Compared to Odyssey's Table Tennis, Pong also offers game management and on-screen score display. Obviously Magnavox sues Atari for patent infringement on the console and wins the lawsuit: Atari, in order not to give up Pong, which is selling very well, agrees to pay a license to continue using the Table Tennis game scheme. Pong was so successful that other companies decided to copy Atari's game, including Bally Midway, Williams Electronics, Sears, and others, all sued by Magnavox. The lawsuits continue for many years and reach all companies that market a "racket and ball" style game, such as Coleco, Mattel, Activision and Nintendo. It is estimated that all of the lawsuits filed until the patents were extinguished in the early 1990s brought about $100 million into the coffers of Magnavox and Sanders.

Derived versions

The console is replaced in 1975 with the Odyssey 100, an economic version of the first model based on an integrated specially made by Texas Instruments to enclose within it several discrete components, and thus reduce costs. The case is also revised, with a more compact plastic shell that also includes controls for playing games. This model offers only 2 games, Tennis and Hockey. Along with it is also presented the Odyssey 200, a version with

3 games (Squash is added) and a rudimentary scoring system based on a progress bar that grows as the player scores a point (fig. 9). The console is erroneously indicated as being capable of handling 2 or 4 players because the "rackets" on the screen can be doubled but, in reality, the game controls are for two players only.

In 1976 three new models arrived. The Odyssey 300, presented to compete with the Coleco Telstar, of which it takes the integrated on which it is based, offers the same 3 games of the 200 model even if, using a chip specifically developed for games, they have a playability and a yield on the screen different from the previous Odyssey and almost identical to those of the Pong clones. The Odyssey 400, an evolved version of the 200 with the same games but with an on-screen scoring system and automatic "batting". The Odyssey 500, which represents the true evolution of the series. It offers color graphics and 3 sprites that replace the rackets in the 3 integrated games: a tennis player, a field hockey player and a squash player, which appear in the corresponding games.

Between late 1976 and early 1977 comes the Odyssey 4305. It is a Magnavox color TV with an Odyssey 300 or 500 integrated. In 1977 comes the Odyssey 2000, an improved version of the 300 model. The games are the same but there is a numerical scoring system that appears on the screen. Also in 1977 came the last two models in the series, the Odyssey 3000 and Odyssey 4000. The first is a variant of the 2000 but with a more modern style while the 4000 offers a still different design, color graphics and joysticks again separated from the machine body.

Also worth mentioning are the Philips Odyssey 200, 2001 and 2100 consoles: they are not clones but original consoles sold under the brand of the Dutch multinational company due to the fact that the latter bought Magnavox in 1974. The 200 is identical to its American counterpart while the 2001 is a 4000 with a specific container and the 2100 is a different version with a different integrated.

The Odyssey line was replaced in 1978 by the Magnavox Odyssey 2, a second generation console that dominated the video game market along with the Atari 2600, ColoecoVision and Intellivision. But that's another story....

Conclusions

The Magnavox Odyssey was a true pioneer. It pioneered home video games, making it possible to use the TV screen for activities other than just watching TV broadcasts. But the Odyssey is more than that: it wasn't just an "electronic game" for kids, it was also for adults. Or at least that's how Magnavox wanted the console to be viewed. Why? Because the not indifferent purchase price was not an amount that parents could spend light-heartedly for a simple toy. Magnavox therefore pushed the fact that the fun offered by the console was also enjoyable by adults, so as to justify the purchase for a use of the console made by the whole family. And in fact in many advertisements the controllers always appear in the hands of adults.

Then, universally the Odyssey is considered a console that did not have a good commercial success but this does not correspond to reality. If we consider the period in which it was released and the fact that it practically created a new sector, the 350,000 units sold are instead an excellent result. It must be considered that the sales have always gone to grow in the 3 years in which it remained on sale: even, for the festivities of 1973 Magnavox had to postpone in production the console because the units that it had estimated to sell had already been all purchased.

Finally, the games offered by the Odyssey should not be judged with the eyes of a videogamer of the 80's because a white square bouncing around the screen can't hold a candle to the graphics of the games we were used to with the home computers and consoles of the following decade, but they still entertained thousands of kids, and that is the ultimate goal of a videogame. Table Tennis/Pong then became so famous that it was offered on all the systems that followed over the years and is still present in the stores of our mobile devices. If this isn't a success...

Optimize PI calculation with Monte Carlo method

by Carlo Luciano Bianco

These pages are inspired by Marco Pistorio's interesting article "Pi, Monte Carlo and Random Numbers", which appeared in RMW issue #09-EN. If you have not already done so, I invite you to read the article of Marco before continuing, since in the following, not to dwell too much, I will take for granted its content. In particular, in these pages I would like to deepen possible ways to speed up the execution of the program in BASIC V2 (without graphics) that has been presented by Marco in his article, and that I report here below for simplicity:

```
10 CI=0:REM INTERNAL POINTS COUNTER
20 NM=1000: REM TOTAL NUMBER OF POINTS
30 FOR PP=0 TO NM
40 XP=RND(1)
50 YP=RND(1)
60 IF XP<=SQR(1-YP*YP) THEN CI=CI+1
70 NEXT
80 PRINT "PI (APPROXIMATE)=";(4*CI/PP)
90 END
```

Obviously, first of all it must be told that the best way to speed up this calculation would be to assemble a program in machine language instead of BASIC V2, no doubt about it. But it is interesting to note that, even in the context of the slowness of BASIC V2, you can still do some small code optimization to try to speed things up a bit.

The first thing we need is a reliable system to measure the execution time of the program. We could surely use a clock or a stopwatch, but this system is not very reliable. In fact, in our specific case, we would be forced to stare at the screen for several minutes without being able to distract ourselves while waiting for the end of the program execution, and there would always be the risk of someone calling or knocking at the door at the wrong time and making us miss the crucial moment. Fortunately, our Commodore 64 is equipped with an internal clock, which perhaps will not have the precision of one millionth of a second, but for the purpose we are interested in, it is absolutely perfect. Moreover, if we use the internal clock of the Commodore 64 to measure the execution time, we can easily use the Turbo and/or Warp functions of any emulator to execute the program saving us many minutes of waiting, since in this case also the internal clock will be speeded up and therefore its measurement of the execution time will be correct.

The internal clock of the Commodore 64 can be manipulated by BASIC V2 using the special variable TI$. If in fact we try to execute the command

```
PRINT TI$
```

we get something like "000241", which represents, in HHMMSS format, the time that has passed since the last clock reset (typically since the machine was turned on), in this case 00 hours, 02 minutes, and 41 seconds. If we then type the command:

```
TI$="000000"
```

this will reset the internal clock, which then starts counting again from 0.
So at this point the method of measuring the execution time should be clear: we add as the first instruction of the program the one that resets the internal clock, and as the last instruction a PRINT TI$. So each time we will know, in HHMMSS format, the time it took to execute the program. The new listing will then be:

```
5 TI$="000000": REM RESET INTERNAL CLOCK
10 CI=0:REM INTERNAL POINTS COUNTER
20 NM=1000: REM TOTAL NUMBER OF POINTS
30 FOR PP=0 TO NM
40 XP=RND(1)
50 YP=RND(1)
60 IF XP<=SQR(1-YP*YP) THEN CI=CI+1
70 NEXT
80 PRINT "PI (APPROXIMATE)=";(4*CI/PP)
85 PRINT "TIME SPENT: ";TI$
90 END
```

in which we have inserted lines 5 and 85 (thus without

renumbering the others).
Running the program now, we find that the time taken is 000115, or 00 hours, 01 minutes, and 15 seconds. In other words, 75 seconds. Changing the value of NM on line 20, we obtain that, for example, with NM=100 the time taken is 7 seconds and with NM=10000 the time taken is 12 minutes and 39 seconds (i.e., 759 seconds).

From now on we will focus on the case NM=1000, both



because it is the case of the original program, and because it corresponds to a time that is long enough for us to clearly see the effects of the changes we are going to make to the code, and at the same time short enough that we don't have to wait hours and hours to do a reasonable number of tests.

If we read the program carefully, we see that lines 5, 10, 20, 80, 85, 90 are each executed once, so they surely have a negligible impact on the execution time of the program, while lines 30 to 70 correspond to a FOR/NEXT loop that is executed NM times. Since the execution time of the program depends on NM (from the test we did a while ago we can clearly see that multiplying or dividing NM by 10 also multiplies or divides the execution time by 10), the part we need to speed up is surely this FOR/NEXT loop.

Let's start analyzing the FOR/NEXT loop in detail, and see what changes we can make to make it faster:
- Lines 30 and 70 are the ones that define the loop, increment the PP variable, check its value against NM, and decide whether to repeat the loop or terminate its execution. There is not much that can be done here to speed up the execution.
- Lines 40 and 50 are the ones that draw the two random numbers. Again, to a first approximation there is not much we can do. If we wanted to, we could look for a faster way of extracting the random numbers than the RND function, but we'll leave that aside for now (of course any suggestions on faster methods of extracting a random number are welcome!).
- That leaves line 60, which is actually the most interesting one for our purpose. This line is quite complex, and contains within it three different operations:
1) Calculation of the function SQR(1-YP*YP);
2) Comparison of the result with the XP value;
3) based on the result of the comparison, increase or decrease in the CI variable.

Incrementing the CI variable is an elementary instruction (an addition), and there is no way to execute it faster. The comparison (IF/THEN) is in fact a quite complex instruction to execute, and requires several machine cycles, but the algorithm we used to calculate Pi necessarily involves a comparison to understand if the point is inside or outside the circle and therefore we can not in any way avoid that IF/THEN. This leaves only the calculation of the function SQR(1-YP*YP) as a candidate for possible speedup. In fact, the SQR function (like many other complex mathematical functions) in BASIC V2 takes a long time to calculate, and calculating it NM times surely has a strong impact on the execution time of the program. Is there any way to perform this calculation faster? Actually yes, and what's more we have several possibilities in front of us.

The first possibility we are going to explore is to ask whether it is really necessary to calculate that SQR function. To get the answer, let's go and reread Marco's article on RMW #09-EN, where the calculation algorithm is described. There we find written that the circumference formula in our case reduces to:

```
X^2=1-Y^2
```

which we can then rewrite as X=SQR(1-Y^2) thus obtaining the three situations X<=>SQR(1-Y^2). However, since we are working with positive numbers, comparing two numbers or their squares is exactly the same thing:
if X=SQR(1-Y^2) necessarily X^2=1-Y^2, and vice versa;
if X<SQR(1-Y^2) necessarily X^2<(1-Y^2), and vice versa;
if X>SQR(1-Y^2) necessarily X^2>(1-Y^2), and vice versa.
In other words, comparing X with SQR(1-Y^2) or X^2 with (1-Y^2), for our purpose, is exactly the same thing. The only difference between the two situations is that in the former we have to calculate the SQR function, while in the latter we can avoid this at the price of calculating

X^2. We expect X^2, which is a simple multiplication, to be faster to calculate than the SQR function, and so we try to modify the program accordingly. Line 60 will then become:

```
60 IF XP*XP<=(1-YP*YP) THEN CI=CI+1
```

Compared to the previous version, the SQR function no longer appears and the XP variable is now multiplied by itself. Our program has therefore become:

```
5 TI$="000000": REM RESET INTERNAL CLOCK
10 CI=0:REM INTERNAL POINTS COUNTER
20 NM=1000: REM TOTAL NUMBER OF POINTS
30 FOR PP=0 TO NM
40 XP=RND(1)
50 YP=RND(1)
60 IF XP*XP<=(1-YP*YP) THEN CI=CI+1
70 NEXT
80 PRINT "PI (APPROXIMATE)=";(4*CI/PP)
85 PRINT "TIME SPENT: ";TI$
90 END
```

Now let's try to run it, and we can see that the time taken is 000027, that is 27 seconds. We used NM=1000, so we must compare this time with the 75 seconds that the program took with SQR, and we see that the time is reduced to less than half. If we use NM=10000, we have that the execution time is 000433, so 273 seconds compared to 759 seconds before.

We might now ask whether this modification, which sped up the execution of the program, also has any effect on the result. To make sure that the modified program is exactly equivalent to the original one, we can calculate pi in the two different ways and compare the result. So let's restore the old line 60 and transform the new line 60 into line 65, with some minor modifications:

```
5 TI$="000000": REM RESET INTERNAL CLOCK
10 CI=0:REM INTERNAL POINTS COUNTER
15 CT=0
20 NM=1000: REM TOTAL NUMBER OF POINTS
30 FOR PP=0 TO NM
40 XP=RND(1)
50 YP=RND(1)
60 IF XP<=SQR(1-YP*YP) THEN CI=CI+1
65 IF XP*XP<=(1-YP*YP) THEN CT=CT+1
70 NEXT
80 PRINT "PI (APPROXIMATE 1)=";(4*CI/PP)
83 PRINT "PI (APPROXIMATE 2)=";(4*CT/PP)
85 PRINT "TIME SPENT: ";TI$
90 END
```

The new line 65 uses CT and not CI as the variable to increment, so that the two calculations are independent. We then added line 15 to reset CT to zero and line 83 to print the new result. Running it, we obtain that the two values ("approximate 1" and "approximate 2") are absolutely identical to each other up to the last decimal place. We have therefore verified that the modified program, which calculates X^2 instead of the SQR function, is absolutely equivalent to the previous one but takes less than half the time to obtain the same and identical result.

As we said above, this is only one of the possible modifications of line 60 that allow to speed up the execution of the program. Another possibility is to make use of the Taylor series that Marco mentioned in his article "RetroMath" on RMW #10-EN. In that article Marco developed in Taylor polynomials the sine and cosine functions, but nothing prevents you from developing also the square root function. For a more detailed discussion I refer you to Marco's article and the references he cited, as far as we are interested here is enough to say that:

```
SQR(1-Y^2)=1-(Y^2)/2+...
```

where we have written explicitly only the terms up to the second order of the Taylor polynomial. We can be pretty sure that calculating [1-(Y^2)/2] is much faster than calculating SQR(1-Y^2), since we avoid calculating the SQR function at the cost of simply dividing by 2 extra. So we could rewrite line 60 as:

```
60 IF XP<=1-YP*YP/2 THEN CI=CI+1
```

So our program will now be:

```
5 TI$="000000": REM RESET INTERNAL CLOCK
10 CI=0:REM INTERNAL POINTS COUNTER
20 NM=1000: REM TOTAL NUMBER OF POINTS
30 FOR PP=0 TO NM
40 XP=RND(1)
50 YP=RND(1)
60 IF XP<=1-YP*YP/2 THEN CI=CI+1
70 NEXT
80 PRINT "PI (APPROXIMATE)=";(4*CI/PP)
85 PRINT "TIME SPENT: ";TI$
```

```
90 END
```

Now let's try to execute it, and we find that the time required for execution with NM=1000 is 000027, that is 27 seconds, exactly as before. If we try with NM=10000, we get 000436, i.e. 276 seconds, again almost identical to the 273 seconds before. The two different optimizations (switching to squares or using the Taylor polynomial truncated to second order) seem to be essentially equivalent from the point of view of improving the execution time, but there is an important difference. Let's now try to verify the result as done before:

```
5 TI$="000000": REM RESET INTERNAL CLOCK
10 CI=0:REM INTERNAL POINTS COUNTER
15 CT=0
18 CX=0
20 NM=1000: REM TOTAL NUMBER OF POINTS
30 FOR PP=0 TO NM
40 XP=RND(1)
50 YP=RND(1)
60 IF XP<=SQR(1-YP*YP) THEN CI=CI+1
65 IF XP*XP<=(1-YP*YP) THEN CT=CT+1
68 IF XP<=(1-(YP*YP)/2) THEN CX=CX+1
70 NEXT
80 PRINT "PI (APPROXIMATE 1)=";(4*CI/PP)
83 PRINT "PI (APPROXIMATE 2)=";(4*CT/PP)
84 PRINT "PI (APPROXIMATE 3)=";(4*CX/PP)
85 PRINT "TIME SPENT: ";TI$
90 END
```

Compared to before, in this case we have further added line 68, which calculates using the Taylor polynomial of the root truncated to the second order and increments the variable CX, and lines 18 to zero CX and 84 to print the result. We can see that the values "approximate 1" and "approximate 2" are absolutely identical up to the last decimal place, as they were before, while the value "approximate 3" is very different. This is because we truncated the Taylor polynomial to the second order, and so what we calculate is not equivalent to before. It is not true, in general, that if X<(1-(YP*YP)/2) then X<SQR(1-YP*YP), and the same is true in all other cases. So, by substituting the square root for its second-order truncated Taylor polynomial, we have obtained a speedup virtually identical to the previous one, but at the price of having a different and, in general, less precise result. Obviously things would improve if we used more terms in the Taylor polynomial, but at that point the time required for the calculation would also be greater.

In conclusion, the best optimization strategy in this case



is to switch to comparing squares as in the first case. The result is absolutely identical to the original program and the time taken is less than half as long.

However, there are cases in which replacing a function with its truncated Taylor polynomial not only does not worsen the final result, but actually, paradoxically, makes it more accurate.

We will see this in a future article, so please don't miss the next issues of RMW.

Programming under GEOS with BeckerBASIC

by Francesco Fiorentini

In November, looking for some material for our **RetroLiPS** project, I came across a game in Basic for ZX81 entitled **Planet Raith**. After having transcribed the code and shared it on our site, I played it a bit, finding it catchy even in its simplicity.
After a few games, seeing the immediacy of the code, I decided to make a porting for Commodore 64, to be published on our website. The porting was quite easy and did not require more than one evening of work.

As usual, I shared both works on some Facebook groups to increase the visibility of these retrievals, but also to encourage other users to do the same: type programs taken from old magazines, make a porting of those we have already shared, or simply share their work with us...

Generally these posts always get a good reception: someone remembers to have typed, in the past, the same listings or to have played a particular game. Once again, the comments I received this time were all pretty interesting. Among all of them, a couple have struck me in particular. In fact, it seems that the game in question is not a completely original game, but it is a clone of a game that already existed.

In the specific case of Planet Raith, which I thought was a game originally proposed by the English magazine **Popular Computing Weekly of April 29, 1982 Vol. 1 No. 2**, it is instead a clone of a famous game written many years earlier: **King of Sumeria**.

This game seems to have created a myriad of clones for the most varied home computers of the eighties: some set in the past, like the original game, others set in the future, or in space, as in the case of Planet Raith.

```
HAMURABI:

LAST YEAR
=     0 STARVED,
=     5 ARRIVED,
POPULATION IS=    100

THE CITY OWNS=   1000 ACRES.

WE HARVESTED =     3 BUSHELS PER ACRE;
 RATS ATE =    200 BUSHELS,YOU NOW HAVE
=  2800 BUSHELLS IN STORE.

HAMURABI:  LAND IS TRADING AT =    21 BUSHELS PER ACRE;
HOW MANY ACRES OF LAND DO YOU WISH TO BUY?
:50

BUSHELS TO USE AS FOOD?
:1000

HOW MANY ACRES OF LAND DO YOU WISH TO
PLANT WITH SEED? :500
```

Fig. 1 - The game King of Sumeria

King of Sumeria is a game written in **1968** on **PDP-8** using the **FOCAL** language (an ancestor of Basic) by **Doug Dyment**. Some people remembered it as **Hamurabi**, because that's the first text you see when the game is launched (see figure 1).

The fun facts are that King of Sumeria is itself a clone of another game, **The Sumerian Game**, known as the first narrative video game, and that the first video game writer was a woman named **Mabel Addis**.

For those who want to learn more about this history, I recommend reading this article:
https://www.acriticalhit.com/sumerian-game-most-important-video-game-youve-never-heard/

Personally I didn't know the story of The Sumerian Game, but after reading that article, the curiosity for the FOCAL language and the PDP-8 system grew in me... Stay tuned on this channel if you want to know more, there will be surprises...

Programming under GEOS

After a necessary introduction, we come to the crucial part of our article. **GEOS** is the Holy Grail of the Commodore 64, a windowed, mouse-controlled, high-resolution operating system that has always tickled my fantasy and ever since the first time I got my hands on it, I've wanted to do something with it. But how?

GEOS programming is not easy and in the eighties finding tools and information to make something with it was absolutely out of my reach. I had to keep this desire closed in a drawer... At least until now, when information and tools are now available to everyone.

Let's get one thing clear, despite the vast availability of information, programming in GEOS remains complex and, if you want to get good results, you have to use low-level tools such as geoProgrammer. **geoProgrammer**, is a suite that includes **geoAssembler**, **geoLinker**, and **geoDebugger**; it is the assembly language environment for use with GEOS.

Fortunately over time have been developed a series of

high-level tools that, at least in the intentions of the manufacturers, should have allowed a wider audience of developers to create software for this operating system. Unfortunately, as we shall see, the results are definitely not at the level of geoProgrammer...

geoBasic: this is an unfinished project by Berkeley Softworks and released by RUN magazine. Unfortunately it is riddled with a number of bugs that seriously compromise its use. I tried it in the past and, although I managed to run my program, a subsequent modification containing code taken from the manual blew the whole job.
After a few days of testing, I decided that I could make better use of my time. Now I read on the Lyonlabs site that there is a file by George Wells that lists the known bugs and possible solutions... I might give it another chance in the light of this new information.

geoCom: a German product by Falk Rehwagen. It's a real compiler (it produces machine language output) with a BASIC-like syntax. It looks promising, but I haven't had the chance to try it yet: unfortunately time is limited for everyone.

BeckerBASIC: object of our article, it is a Basic extension produced by Abacus that allows to create programs for GEOS. I won't add anything else, let's see it together.

It is also possible to use **Forth** through two implementations of this language: **geoForth**, by Nick Vrtis and **Brian**, by Hank Wilkinson.

BeckerBASIC

BeckerBASIC is an extension of Basic 2.0 produced by Abacus in 1988 and, according to Abacus, completely compatible with GEOS. Honestly, calling it an extension is a bit reductive, since it is a complete development suite that includes an editor, a test environment and a converter.

BeckerBASIC contains a total of 273 commands that greatly enrich the standard Basic V2 package. Some of these commands can be used to generate standard Basic code, but the part we are interested in is the one that gives us access to the GEOS graphical environment and allows us to exploit its capabilities... Or at least, if not all of them, some of them.

Before delving into the GEOS commands, it is good to say a few words about the compatibility with Basic V2 that, unlike the vast majority of C64 Basic extensions, is not 100% guaranteed. Some commands are in fact redefined

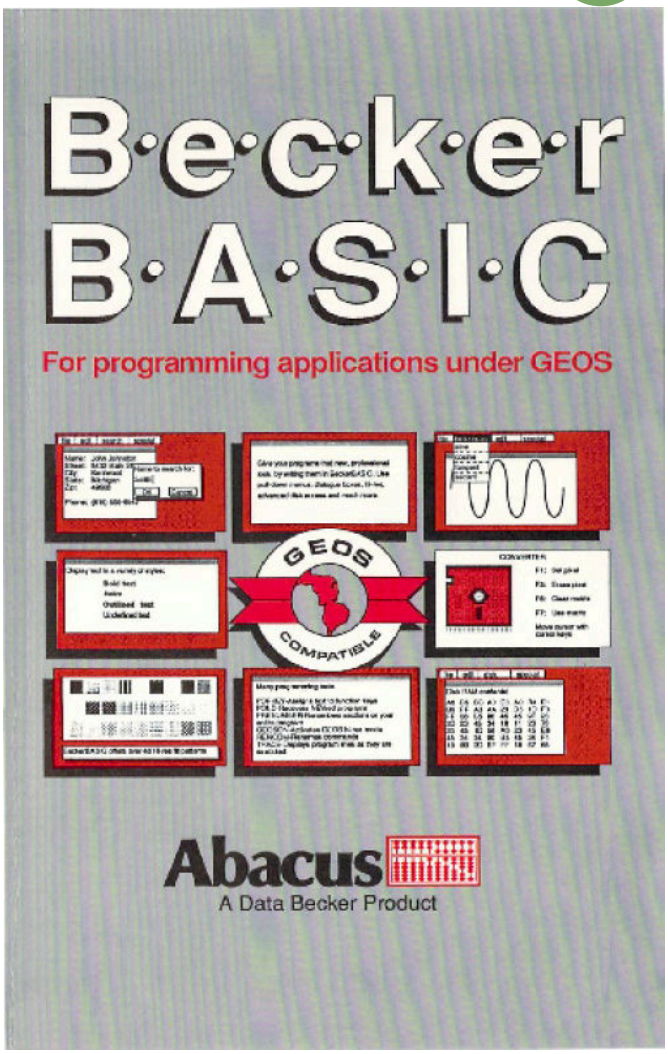


Fig. 2 - The cover of the BeckerBASIC manual

and, using them as we would do with standard Basic, would return an error.

This is the case for the IF command that in BeckerBASIC has this syntax:

```
10 IF <condition> THEN <instructions>
20 ELSE <instructions>
30 ENDIF
```

In BeckerBASIC the IF command has also implemented the ELSE construct and must be terminated by the ENDIF construct, otherwise an interpretation error may occur. A normal line containing IF of a standard Basic program, to be interpreted by BeckerBASIC, must be modified as follows:
Basic V2 -> 10 IF a=0 THEN a=a+1
Becker Basic -> 10 IF a=0 THEN a=a+1:ENDIF

According to the manual it is possible to nest up to 255 IF cycles.
Along with the extended IF command (which we have just seen) and a special command for multiple choice (SELECT), BeckerBASIC offers three new types of loops that allow

more flexible programming than FOR/NEXT: WHILE/DO/ENDDO, REPEAT/UNTIL and LOOP/LPEXTTIF/ENDLOOP.

Select Case

```
10 SELECT <variable>
20 CASES 2,4,6/10:GOSUB "SUBROUTINE1"
30 CASES 1,3,5:GOSUB "SUBROUTINE2"
40 OTHER GOSUB "SUBROUTINE3"
50 ENDSEL
60 END
70 "SUBROUTINE1":PRINT "PIPPO":RETURN
80 "SUBROUTINE2":PRINT "PLUTO":RETURN
90 "SUBROUTINE3":PRINT "PAPERINO":RETURN
```

Note that it is possible to identify a routine and then recall it through GOSUB also using a label, provided that:
1) it is enclosed in quotation marks (")
2) it starts a program line
3) a colon ":" separates the label from the rest of the program line

While...Enddo

```
10 WHILE <condition> DO
20 <instruction>
25 <instruction>
30 ENDDO
```

A WHILE loop is executed as long as a condition remains true and/or the commands within the loop do not change it. You can have up to 15 nested WEND loops.

Repeat...Until

```
10 REPEAT
20 <instruction>
25 <instruction>
30 UNTIL <condition>
```

Unlike the While loop, the Repeat/Until checks the condition at the end. This means that the instructions within the loop will be executed at least once. As with the While loop, you can have up to 15 nested Repeat loops.

Loop...EndLoop

Program instructions between LOOP and ENDLOOP are executed as long as the LPEXITIF condition is met. I take from the manual: "LOOP/ENDLOOP allows common types of loops to be set up. The branch can be designated at any time, which makes LOOP/ENDLOOP extremely flexible and useful when no other loop type can do the job."

```
10 LOOP
20 <instruction>
30 LPEXITIF <condition>
40 . <instruction>
50 ENDLOOP
```

Personally I don't find it very useful, but for the sake of the record I have reported it. As with the other loops, you can have up to 15 nested Loop loops.

I wanted to mention these constructs because, in my humble opinion, they are able to increase the readability of a Basic program and therefore could be useful to our readers.

However, the number of commands added by BeckerBASIC is impressive and it would be impossible to cover them all in one article. For those who want to know more, I invite you to read the manual which, although not always immediate, will be able to shed light on all the commands available.

GEOS Programming with BeckerBASIC

Before we start talking about BeckerBASIC and its ability to program in a GEOS environment, it is good to make a premise: programs written with BeckerBASIC only look like GEOS applications; they do not really work like GEOS applications. They will always need the runtime that comes with the suite to work. If you want to distribute an application developed to run in the GEOS environment, you will have to distribute the Run-Only-System along with it. The programs converted with the Converter, in fact, access the Run-Only-System when you double click on their icon in the deskTop and it is therefore perfectly legal to distribute the Run-Only-System with them (as also explained in the manual).

Actually if you look at it from the side of a purist, we are not programming in GEOS, but using a shortcut to something that can run in HiRes mode within this operating



Fig. 3 - BeckerBASIC's disk content

system. Not all GEOS features are accessible, but the result is quite acceptable and it's fun to learn how to do it. :-)

Second premise: for obvious reasons of space, I will assume that you know how to launch GEOS and use it to run applications. At the end of the article I will leave a small guide that will help you to get the GEOS, launch it and configure the VICE emulator on Windows to use the mosue.

After these premises, let's see how to write a GEOS program with BeckerBASIC! (It's about time... ndR).

First of all we have to run the program to type the listing. This is contained in the BeckerBASIC diskette and it is called System 1, the icon with the label INPUT (see figure 3). As you can see, you only have 15750 bytes available, because the rest of the memory is occupied by GEOS and the BeckerBASIC interpreter.

Below you will find a series of useful commands to manage your program:
- dcsaveb "program_name" -> delete the file and save the program again
- dloadb "program_name" -> load the program
- desktop -> exit the editor and return to the GEOS desktop
- CTRL+commodore -> launch the test environment (you will need it a lot)

Here we have all the 273 BeckerBASIC commands, but we are mainly interested in those that allow us to work in high resolution; let's see all those I used in my program .

GEOSON: this command activates the high resolution graphics of GEOS. This instruction must be invoked before executing any other GEOS command. If you try to use other high resolution (HR) commands without first running GEOSON, you will get an error.

GEOSOFF: disables high resolution.

HRDEL: cleans the screen in high resolution.

HRPRINT: writes text in high resolution
`HRPRINT x,y,tx$`
x,y: are the x and y coordinates of the screen: x goes from 0 to 319, y from 0 to 199
tx$: is the string to be printed on the screen.
ATTENTION: every string sent to GEOS must be terminated with a 0 (CHR$(0))!

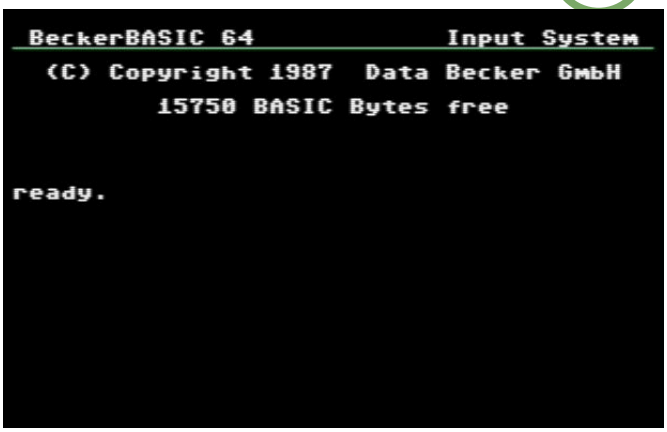


Fig. 4 - Input System of BeckerBASIC (System 1)

Strings can be combined with this set of control characters:
CHR$(08) deletes the last displayed character
CHR$(09) moves the text cursor one character to the right
CHR$(10) moves the text cursor down one character
CHR$(11) moves the text cursor to the starting position (X=0,Y=0)
CHR$(12) moves the text cursor upwards by one character
CHR$(13) moves the text cursor to the beginning of the next line of text following a carriage return
CHR$(14) activates the underline mode. All following characters are underlined
CHR$(15) disables underlining mode
CHR$(18) enables reverse mode. All following characters appear in reverse
CHR$(19) disables reverse mode
CHR$(24) enables bold mode. All following characters appear in bold
CHR$(25) enables italic mode. All the following characters appear in italic style
CHR$(26) enables outline mode. All following characters appear in outline style
CHR$(27) disables all typographical styles (italic, bold, etc.).
Refer to the program for a number of examples of using HRPRINT.

I couldn't find any command to clean a line, so I used a trick in the program. I created an empty string with a terminator: bl$=""+chr$(0) and then printed it over the line I wanted to clean up.

HRGET: reads an input value in high resolution
`HRGET x,y,x$`
x,y: are the x and y coordinates of the screen
x$: is the variable that receives the text. The size of the string must be defined beforehand and must be terminated by chr$(0).

I imagine that it would be natural to define the variable as: x$=""+chr$(0) and it would be obviously correct,

only in this way the input cursor would be positioned at the end of the string (on the right) and to insert the text we would have to move to the left...
To overcome this inconvenience we must define the variable as: x$=chr$(0)+" "
In this way the cursor will be positioned to the left and we can write naturally.

ATTENTION: as you will notice in the program, the HRGET function is slightly offset from the HRPRINT function. If you want to place text and input in the same line, you will have to place HRGET a few pixels higher (about 7).

```
88 hrprint 20,90,chr$(27)+">>>"+chr$(0)
89 x$=chr$(0)+" "
90 hrget 37,83,x$
```

WAITKEYA: waits for a key to be pressed.

DESKTOP: is the equivalent of END. It terminates the program and returns to the GEOS desktop.

These commands would have been enough to realize the porting of the game Planet Raith to GEOS, given its simplicity, but since I had a graphical environment at my disposale and I was in time for my article, I wanted to add a sinmple frame and an animated sprite. The creation of the sprite and its animation were the most fun part of the project. :-)

Drawing the frame was really easy, BeckerBASIC provides a number of graphical commands that I could have used to draw in HiRes, including: HRLINE, HRVLINE, HRHLINE and HRFRAME. Since I did not have particular performance needs, I preferred to use the slower but always good: HRLINE.

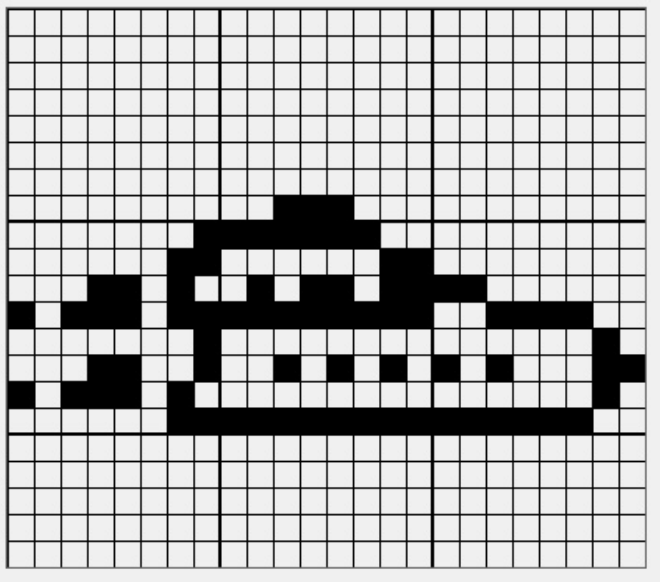


Fig. 5 - The sprite drawn with CBM Prg Studio

HRLINE: draws a line between two points in high resolution.

```
HRLINE x1, y1, x2, y2 [,p]
```

x1,y1: are the coordinates of the starting point
x2,y2: are the coordinates of the arrival point
p: indicates whether pixels are on or off: 0 pixels are on, 1 pixels are off. The default is 0.

The sprite part is a bit more complicated. Even if BeckerBASIC provides a series of commands to manage the sprites in high resolution, the manual, although complete, does not clarify some information and I had to go a bit 'by trial and error'...

MBDESIGN: inserts a string containing sprite data into memory as a 64-byte memory segment.

```
MBDESIGN bl, d$
```

bl: is the memory block where the sprite will be loaded. The value is calculated with the formula bl = memory_address/64. The values of BL range from 0 to 1023. Of course, you cannot use all memory locations; avoid BASIC memory and zeropage memory. WARNING: the value of bl must be an exact multiple of 64.
d$: is a 63-character string containing information for drawing the sprite. Sprites have horizontal resolutions of 24 pixels and vertical resolutions of 21 pixels. A pixel is set (on) or not set (off) according to the bytes that make up the sprite matrix. Each line of 24 pixels occupies three bytes, while the 21 columns occupy 63 bytes. That's why the string is 63 characters long.

I used the sprite editor of CBM Prg Studio to draw the sprite. Unfortunately I'm not graphically gifted so the result is what it is, but technically it is functional. Once finished the drawing I exported it as DATA values and I added this piece of code in my program.

```
10000 'sprite data
10010 data 0,0,0,0,0,0,0
10020 data 0,0,0,0,0,0
10030 data 0,0,0,0,0,0,56,0
10040 date 1,252,0,3,3,0,26,91
10050 date 192,187,255,60,1,0,2,25
10060 date 42,163,186,0,2,3,255,252
10070 date 0,0,0,0,0,0,0,0,0
10080 date 0,0,0,0,0,0,0
```

After that I wrote a FOR loop to load the DATA values into the variable d$ - string of 63 characters:

```
10085 for e=1 to 63
10090 read sp: d$=d$+chr$(sp)
10095 next e
```

I had my sprite, but I had to load it in memory and above all, in a correct memory area to be displayed in high resolution. In fact, although the manual clearly states that the sprite data must be placed in the same 16K memory range as the active screen, in my early tests I had completely ignored this warning. So if we want to draw a sprite in the high resolution screen, its data must be put in the second memory segment, as indicated by the table below:

MEMORY SEGMENT I	MEMORY SEGMENT II
0 - 16383	32768 - 49151
13(832) - 15(960)	552(35328) - 559(35776)

I opted for memory address 552: mbdesign 552,d$

MBBLOCK: associates a block of memory to a sprite (from 1 to 8) and places it in the screen

```
MBBLOCK sc,nr,bl
```

sc: is the screen where you want your sprite to appear. If you want it in low resolution, set sc to 1. If you want the sprite on the screen in high resolution, set sc to 35.
nr: is the number of sprites to which to assign the data block that we loaded in memory with MBDESIGN. Obviously, NR values range from 1 to 8.
bl: is the number of the data block that contains the sprite data. The same value used for MBDESIGN

During my first tests I had mistakenly set nr to 1 and obviously the sprite did not appear in the introductory screen of Planet Raith, but was visible when the game ended ... :-)
Attention, our sprite is not yet visible on the screen. It must be positioned and turned on. Before turning it on, however, we can also decide its color. Let's see how to accomplish this!

MBSETCOL: defines the color of the sprite

```
MBSETCOL nr,cl
```

nr: is the number of sprites
cl: is the color to use. We can use all 16 colors of the C64. The values range from 0 to 15.

MBSETPOS: inserts a sprite at the given position

```
MBSETPOS nr,x,y
```

x,y: are the x and y coordinates of the screen
nr: is the number of sprites

MBON: turns on the sprite

```
MBON nr
```

nr: is the number of sprites

Yay, a black sprite has appeared on the screen in high resolution! How about moving it around a bit? Moving a sprite is really simple, in practice the movement is nothing else than a continuous repositioning of the sprite through the command: MBSETPOS.

```
2560 for e=50 to 290 step 1
2565 mbsetpos 1,e,150
2600 next e
```

MBALLOFF: removes all active sprites from the screen. Obviously you can also turn off a single sprite with the MBOFF nr command; I used this command for convenience.

Well, at this point I finished the explanation of all the commands I used in my program. The list is deliberately discursive to allow everyone to understand the behavior. If you want to test the listato, copy it and then go to the editing environment (System 1) and paste it through the paste function of VICE (I suggest you to use the WARP mode during the paste). Then press CTRL+commodore to start the test environment and from there run it by pressing F1.

To save the file to disk, use the command dcsaveb "program_name" which deletes the file on disk and saves the program again. The program is still in standard Basic format and therefore not bootable by GEOS. If you want to try to transform your program into a GEOS file, you can use the CONVERTER that comes with BeckerBASIC.

Launch the Converter from the GEOS disk, make sure that your file is on drive 8 and follow the on-screen instructions. You can also create a custom icon to associate with your file, as I did. Don't worry if sometimes the converter fails, the program is not foolproof: try again and you will eventually succeed.

You can download the diskette containing the Raith_GEOS game from here:
- https://retromagazine.net/download/rlips/raith_geos.d64

And here is the listing of Raith_GEOS:

```
10 'init sprite'
11 d$=""
12 gosub 10000
18 geoson
19 bl$="                                                "+chr$(0)
20 'go to intro routine'
21 gosub 2000
30 'init var'
31 p=100:i=5:d=0:s=3650:k=350:m=4000
```



Fig. 6 - The disk I created for Raith_GEOS

```
32 l=1000:y=1:c=4:n=2576:
35 d=0
40 'start core'
41 hrdel
42 hrprint 20,20,chr$(27)+"annual controllers
report for year "+str$(n)+chr$(0)
43 n=n+1
44 hrprint 20,40,chr$(27)+"sir, during the last
year:"+chr$(0)
45 hrprint 20,50,str$(d)+" workers died from
starvation"+chr$(0)
46 hrprint 20,60,chr$(27)+"and "+str$(i)+"
immigrants came to raith"+chr$(0)
50 'klingons'
51 if y>0 then goto 57
52 p=int(p/2):
53 hrprint 20,70,chr$(24)+"klingons attacked our
planet"+chr$(0)
54 hrprint 20,80,chr$(24)+"and half of the people
are dead!"+chr$(0)
55 'info'
57 hrprint 20,90,chr$(27)+"the population is
now"+str$(p)+chr$(0)
58 hrprint 20,100,chr$(27)+"there are "+str$(l)
+" mine shafts"+chr$(0)
59 hrprint 20,110,chr$(27)+"the mines produced
"+str$(m)+" tonnes"+chr$(0)
70 hrprint 20,120,chr$(27)+"of minerals at "+str$
(c)+" tonne/mine"+chr$(0)
71 hrprint 20,130,chr$(27)+"but klingons pirated
"+str$(k)+" tonnes"+chr$(0)
72 hrprint 20,140,chr$(27)+"leaving "+str$(s)+"
tonnes in storage"+chr$(0)
73 hrprint 20,170,chr$(27)+"press any key to
continue..."+chr$(0)
76 waitkeya
79 'new mines'
80 hrdel
81 hrprint 20,20,chr$(27)+"there are "+str$(s)
+" tonnes"+chr$(0)
82 hrprint 20,30,chr$(27)+"of minerals in
stock"+chr$(0)
83 c=int(rnd(1)*10+1): v=c+17
84 hrprint 20,40,chr$(27)+"it currently
costs"+chr$(0)
85 hrprint 20,50,chr$(27)+str$(v)+" tonnes to
open a mine"+chr$(0)
86 hrprint 20,70,chr$(27)+"how many mines do you
want to"+chr$(0)
87 hrprint 20,80,chr$(27)+"build this
year?"+chr$(0)
88 hrprint 20,90,chr$(27)+">>>"+chr$(0)
89 x$=chr$(0)+"     "
90 hrget 37,83,x$
91 x=val(x$)
92 if x=0 then goto 105:endif
93 if v*x<=s then goto 104:endif
94 hrprint 20,100,chr$(27)+"there is only enough
stock to"+chr$(0)
95 hrprint 20,110,chr$(27)+"buy materials
for"+chr$(0)
96 hrprint 20,120,chr$(27)+str$(int(s/v))+"
mines"+chr$(0)
97 waitkeya
98 hrprint 20,100,bl$
99 hrprint 20,110,bl$
101 hrprint 20,120,bl$
102 hrprint 20,88,bl$
103 goto 88
104 l=l+x:s=s-v*x
105 'mine shafts'
106 hrdel
107 hrprint 20,20,chr$(27)+"raith currently
has"+chr$(0)
108 hrprint 20,30,chr$(27)+str$(l)+" mine
shafts"+chr$(0)
109 hrprint 20,40,chr$(27)+"how many mines will
you sell?"+chr$(0)
110 hrprint 20,50,chr$(27)+">>>"+chr$(0)
111 x$=chr$(0)+"     "
112 hrget 37,43,x$
113 x=val(x$)
114 if x=0 then goto 130:endif
115 if x<l then goto 125:endif
116 hrprint 20,70,chr$(27)+"controller, you only
have"+str$(l)+chr$(0)
117 hrprint 20,80,chr$(27)+"mines... try selling
a few less"+chr$(0)
118 waitkeya
```

```
119 hrprint 20,70,bl$
120 hrprint 20,80,bl$
121 hrprint 20,48,bl$
122 goto 110
125 l=l-x: s=s+v*x
130 'selling stock'
131 hrdel
132 hrprint 20,20,chr$(27)+"there are "+str$(s)
+" tonnes"+chr$(0)
133 hrprint 20,30,chr$(27)+"of minerals in
stock"+chr$(0)
137 hrprint 20,50,chr$(27)+"how much stock will
you sell"+chr$(0)
138 hrprint 20,60,chr$(27)+"to feed the hork
force?"+chr$(0)
139 hrprint 20,70,chr$(27)+">>>"+chr$(0)
141 x$=chr$(0)+"     "
142 hrget 37,63,x$
143 f=val(x$)
144 if f<=s then goto 160:endif
145 hrprint 20,90,chr$(27)+"such generosity
commander but"+chr$(0)
146 hrprint 20,100,chr$(27)+"there are only "+str$
(s)+" tonnes"+chr$(0)
147 hrprint 20,110,chr$(27)+"of minerals in
stock..."+chr$(0)
148 waitkeya
150 hrprint 20,90,bl$
151 hrprint 20,100,bl$
152 hrprint 20,110,bl$
153 hrprint 20,68,bl$
154 goto 139
160 s=s-f:d=p-int(f/20):i=0:
162 if d>=0 then goto 165:endif
165 i=-d/2:d=0
170 'mines work'
171 hrdel
172 hrprint 20,20,chr$(27)+"raith has "+str$(l)
+" mines shafts"+chr$(0)
173 hrprint 20,30,chr$(27)+"how man mines will
be worked?"+chr$(0)
175 hrprint 20,40,chr$(27)+">>>"+chr$(0)
176 x$=chr$(0)+"     "
177 hrget 37,33,x$
178 g=val(x$)
180 if g=0 then goto 400:endif
182 if g<=l then goto 195:endif
185 hrprint 20,60,chr$(27)+"imaginative thinking
sir but"+chr$(0)
186 hrprint 20,70,chr$(27)+"we only have "+str$
(l)+" mines shafts"+chr$(0)
188 waitkeya
190 hrprint 20,60,bl$
191 hrprint 20,70,bl$
192 hrprint 20,38,bl$
193 goto 175
195 if int(g/2)<=s then goto 220:endif
200 hrprint 20,60,chr$(27)+"0.5 tonne of stock
has to be"+chr$(0)
202 hrprint 20,70,chr$(27)+"sold to purchase
equipment for"+chr$(0)
204 hrprint 20,80,chr$(27)+"each active mine.
we have enough"+chr$(0)
206 hrprint 20,90,chr$(27)+"stock to operate
"+str$(int(s/2))+" mines"+chr$(0)
208 waitkeya
210 hrprint 20,60,bl$
211 hrprint 20,70,bl$
212 hrprint 20,80,bl$
213 hrprint 20,90,bl$
215 hrprint 20,38,bl$
217 goto 175
220 if g<=10*p then goto 400:endif
221 hrprint 20,60,chr$(27)+"commander, one worker
can"+chr$(0)
222 hrprint 20,70,chr$(27)+"supervise only 10
```





Fig. 7 e 8 - Two frames of the sprite animation in the intro of Planet Raith for GEOS

```
mines. we can"+chr$(0)
224 hrprint 20,80,chr$(27)+"only have a maximum
of "+str$(int(10*p))+chr$(0)
226 hrprint 20,90,chr$(27)+"mines operating at
present"+chr$(0)
228 waitkeya
230 hrprint 20,60,bl$
231 hrprint 20,70,bl$
232 hrprint 20,80,bl$
233 hrprint 20,90,bl$
235 hrprint 20,38,bl$
237 goto 175
400 'computation'
410 c=int(rnd(1)*5+1): m=g*c: k=0:
q=int(rnd(1)*5+1)
411 if int(q/2)<>q/2 then goto 415:endif
412 k=int(s/q)
415 s=s-k+m: q=int(rnd(1)*5+1): i=int(c*(20*l+s)/
p/100+1)
420 p=p+i: q=int(f/20): y=int(10*((rnd(1)*3+1)-.3))
425 if p<q then goto 35:endif
428 d=p-q
430 if d>.5*p then goto 1000:endif
440 if d>.35*p then goto 460:endif
450 if d>.2*p then goto 1200:endif
460 p=q
490 if s>=100*p then goto 1300:endif
495 goto 40
1000 'assassinated'
1001 hrdel
1005 hrprint 20,30,chr$(27)+"you have been
assassinated..."+chr$(0)
1010 hrprint 20,50,chr$(27)+"play again?  1.yes
0.no"+chr$(0)
1011 hrprint 20,60,chr$(27)+">>>"+chr$(0)
1015 x$=chr$(0)+" "
1017 hrget 37,53,x$
1018 r=val(x$)
1020 if r=0 then goto 1800
1025 else goto 30
1030 endif
1200 'rebellion'
1205 hrprint 20,30,chr$(27)+"the workers are
hungry and"+chr$(0)
1210 hrprint 20,40,chr$(27)+"there is talk of
rebellion..."+chr$(0)
1220 waitkeya
1280 goto 460
1300 'another planet'
1301 hrdel
1302 hrprint 20,30,chr$(27)+"you have gone to
```

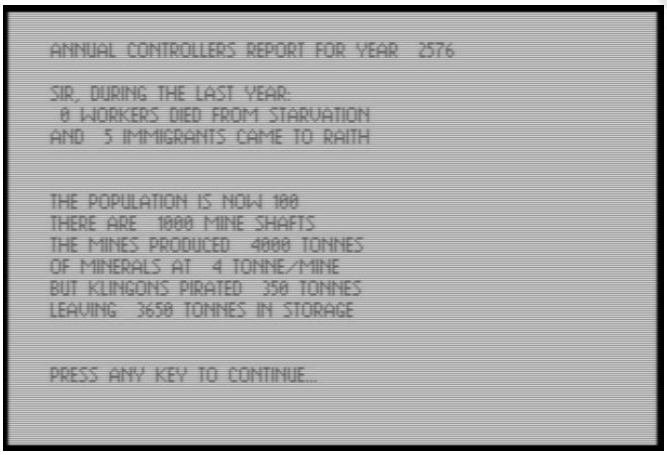


Fig. 9 - Initial phase of the game

```
another planet"+chr$(0)
1303 hrprint 20,40,chr$(27)+"with the contents
of the"+chr$(0)
1304 hrprint 20,50,chr$(27)+"treasury..."+chr$(0)
1310 hrprint 20,70,chr$(27)+"play again?  1.yes
0.no"+chr$(0)
1311 hrprint 20,80,chr$(27)+">>>"+chr$(0)
1315 x$=chr$(0)+" "
1317 hrget 37,73,x$
1318 r=val(x$)
1320 if r=0 then goto 1800
1325 else goto 30
1330 endif
1800 'return to geos after key press'
1801 hrdel
1805  hrprint  110,100,chr$(27)+"...goodbye...  :-
("+chr$(0)
1810 waitkeya
1820 geosoff
1830 desktop
1850 '--- end ---'
2000 'intro routine'
2001 hrdel
2150 hrprint 110,30,chr$(24)+"planet
raith"+chr$(0)
2160 hrprint 80,50,"(c)2021 geos version
by"+chr$(0)
2170 hrprint 90,70,chr$(24)+"francesco
fiorentini"+chr$(0)
2175 hrprint
264,124,chr$(24)+chr$(14)+"raith"+chr$(15)+chr$(0)
2180 hrprint 130,170,chr$(24)+"press any key to
continue"+chr$(0)
2500 'create the sprite and move it'
2510 hrline 10,10,310,10:hrline 10,190,310,190
2515 hrline 10,10,10,190:hrline 310,10,310,190
2550 mbdesign 552,d$
2551 mbblock 35,1,552
```

```
2552 mbsetpos 1,50,150
2553 mbsetcol 1,0
2555 mbon 1
2560 for e=50 to 290 step 1
2565 mbsetpos 1,e,150
2600 next e
2690 waitkeya
2695 mballoff
2900 return
10000 'sprite data and load into string'
10010 data 0,0,0,0,0,0,0,0
10020 data 0,0,0,0,0,0,0,0
10030 data 0,0,0,0,0,0,56,0
10040 data 1,252,0,3,3,0,26,91
10050 data 192,187,255,60,1,0,2,25
10060 data 42,163,186,0,2,3,255,252
10070 data 0,0,0,0,0,0,0,0
10080 data 0,0,0,0,0,0,0,0
10085 for e=1 to 63
10090 read sp: d$=d$+chr$(sp)
10095 next e
10100 return
```

This is just a first taste of GEOS programming, I'm already working on other little projects. Don't miss the next issues of RMW if you are interested in learning more. And, if you found this article to your liking, or if you have any questions or comments, please feel free to let us know through our institutional channels.

GEOS Guide
Short guide to relive the emotions of this fantastic Commodore 64 Operating System.
The guide is written taking into account the Vice emulator on Windows.

Needed:
- Vice emulator: https://vice-emu.sourceforge.io/index.html#download
- DirMaster program: https://style64.org/dirmaster

Download GEOS from here:
- http://www.zimmers.net/anonftp/pub/cbm/geos/c64/GEOS64.ZIP

Before running GEOS make sure you have the 1351 mouse drive in the correct position
- using DirMaster, open the disk GEOS64.D64
- using the arrow keys move the mouse file higher than the joystick file
- save the disk GEOS64.D64 again

To run GEOS:
- insert the floppy disk GEOS64.D64 in VICE
- start GEOS with the command LOAD "GEOS",8,1
- when the GEOS desktop is loaded, you have to enable the mouse
- go to Settings -> Control Port Settings...
- select 'Mouse (1351)' from the dropdown menu under Control Port 1 device and then OK
- and then from Settings -> Mouse Settings -> Grab Mouse events Alt-Q (Alt-M for Vice GTK)
- to regain control of the mouse from Windows, press Alt-Q (Alt-M for Vice GTK) again

Have fun!!!

Useful links

Download BeckerBASIC:
- https://www.lyonlabs.org/commodore/onrequest/geos/becker-basic.d64

BeckerBASIC Manual:
- https://www.lyonlabs.org/commodore/onrequest/geos/Becker_Basic_Manual.pdf

History of The Sumerian Game:
- https://www.acriticalhit.com/sumerian-game-most-important-video-game-youve-never-heard/

Everything you'd like to know about GEOS:
- https://www.lyonlabs.org/commodore/onrequest/geos/index.html

Download the game Raith_GEOS:
- https://retromagazine.net/download/rlips/raith_geos.d64

Redefined characters for C128 (80 columns mode)

by Gianluca Girelli

The problem of font redefinition on 8-bit computers has been addressed several times on these pages since the historical Number 1 of RMW. As is now known, this expedient was necessary not only to add new fonts to the only set available by default on the machines of the time, but also to add a series of original graphic elements to be used for programming videogames or even just to embellish any management software.

Although some home computers of the time had a special command for this task (e.g. the "SYMBOL" of the Locomotive BASIC of the Amstrad CPC), in most cases it was necessary to interact with the system registers through the PEEK and POKE commands. This was, for example, the case with the C64 and the C128 in 40-column mode. The detailed modes have been explored several times in previous issues to which I refer you in case of doubt.

In this article we deal instead with the problem of the redefinition of the characters for the C128 in 80 columns mode. For those who do not know, this mode was added (unlike all previous Commodore models) to orient the computer towards a business user and was made possible by the addition on the motherboard of a special chip, the 8563, totally independent from the VIC (8564) and with its own RAM.

Long time readers (and "geeks" in general) know that the VIC has registers that have the same address of the RAM, that the VIC sees in "pages" of 16KB; that's why reprogramming the characters is, with due "caution", a "simple" exercise of BASIC use with direct access to memory. In the 80-column mode of the C128 this is not possible, since access to the 8563 is allowed only through the use of machine language. In particular, this chip has



Fig. 1 - The Japanese character set

37 internal registers that can be read and written only through two dedicated I/O locations: 54874 ($D600) and 54875 ($D601). The operation is as follows: you must enter in 54874 the number of the register you want to interact with, and when bits 6 and 7 will take the value "1" (managed by the system), you can store a value in 54875. This will force the passage of that value in the register pointed by 54874.

Now let's see how to translate this into Assembly 6502 language. For our purposes, we will limit ourselves in this article only to the part related to writing the registers and memory of the chip, which is necessary to proceed with character redefinition.
We'll explore reading modes in a later article.

```
F0B00       LDY #$00
            STY $FF00
            STA $D600
F0B08       BIT $D600
            BPL $0B08
            STX $D601
            RTS
```

In the next article we will analyze in more detail how the 8563 works, but for now it is enough to know that the write routine has been stored in the cassette buffer (decimal address 2816, hexadecimal 0B00) and will be activated by our code with the call "SYS 2816,A,X". Unlike C64 BASIC it is possible to pass values during the "SYS" call of BASIC 7.0 that will be stored in the registers of processor 6502: in our case we will use the accumulator (register "A") to define in which register of 8563 we want to write and the register "X" to send the value to be written.

The meaning of the LM routine above is as follows:
- the first two lines select bank 15 of the C128 memory (more information can be found in #28);
- the third line loads the contents of the accumulator in the I/O register 54874: the accumulator contains the number of the register of 8563 in which we want to write;
- lines 4 and 5 are a loop that continues until bits 6 and 7 of the register go to "1";
- the sixth line writes, as soon as possible, the content of register "X" in 54875: as a consequence, this value is automatically passed in the register pointed by 54784.



Fig. 2 - The 80 columns of the Commodore 128

As for the 80-column mode, the character description is contained in the RAM of the chip starting from address 8192 ($2000), using for each character 16 bytes: the first 8 contain the description of the single character as for all other Commodore 8-bit computers, while the second 8 are not used and our write routine will therefore have to take them into account.

Among the many registers of the 8563 there are some (three, in particular) extremely interesting for our purposes. They are "18", "19" and "31" and they are used respectively to store a 16 bit address ("high" byte in register 18 and "low" byte in register 19, contrary to what usually happens with 6502 addresses, which is a "little endian" processor) and to store the data to be transcribed at that address. Let's take a concrete example: suppose we want to start defining a character instead of "@" (first character in the alphanumeric sequence of the font).

As we know, each symbol is defined by a matrix of 8*8 pixels and therefore composed of a sequence of 8 bytes. Let's suppose the first byte has value "186" and must be written in location 8192, which in hexadecimal corresponds to $2000. Let's divide the address into the two high and low bytes that make it up ($20 and $00) and let's translate them into decimal ($20=32; $00=0). At this point it will be easy to see what the sequence does:

```
SYS 2816,18,32
SYS 2816.19.00
SYS 2816.31.186
```

the first two calls are used to recompose the 16 bit address where to write; the third one is used to provide the data to write.

An interesting thing about registers "18" and "19" is that they are self-incrementing: after the operation is executed, the address in them automatically increments by 1, allowing fast memory block transfer operations. That's why, in the following code, the initialization of registers "18" and "19" is done only once for each block and not at each iteration of the loop.

As announced, we will explore the functionality of the 80-column chip later. For now, I'll leave you with the full code that deals with the implementation of the first 26 characters of a new "Japanese-like" looking font.

Happy coding!

```
5 color 6,2
6 color 5,3
10 print chr$(147)
11 print "japanese-like set on commodore
128":print
12 print "80-column version":print
13 print "original code by":print
18 print "gianluca girelli, dec 21":print
20 print "**********************************"
21 print:print
22 print"a b c d e f g h i j k l m":print
23 print"n o p q r s t u v w x y z"
30 rem installs assembly routine to write
8563 registers
32 rem inside cassette buffer
40 gosub 410
42 :
50 rem installs new japanesese-like charset
60 gosub 890
70 stop
75 rem resets situation if needed
80 bank 15:sys 52750,0
85 end
410 rem assembly routine to write 8563
registers
420 for i=2816 to 2832:read a$:poke
i,dec(a$):next
430 data a0,00,8c,00,ff,8d,00,d6
440 data 2c,00,d6,10,fb,8e,01,d6,60
450 return
460 :
890 rem writes characters data into 8563 ram
900 for i=1 to 15
905 :  sys 2816,18,32:sys 2816,19,16*i
910 :  for k=1 to 8
915 :     read x
920 :     sys 2816,31,x
925 :  next k
930 next i
935 rem needs to split it un in two parts
936 rem since registers are 8-bit only
940 for i=0 to 10
945 :  sys 2816,18,33:sys 2816,19,16*i
950 :  for k=1 to 8
955 :     read x
960 :     sys 2816,31,x
965 :  next k
970 next i
975 return
1000 data 126,36,36,126,36,36,36,0
1010 data 94,66,66,252,66,66,124,0
1020 data 112,64,64,192,64,64,62,0
1030 data 94,66,66,194,66,66,124,0
1040 data 126,64,64,254,64,64,62,0
1050 data 62,64,64,254,64,64,64,0
1060 data 112,64,64,254,66,66,62,0
1070 data 194,66,66,126,66,66,130,0
1080 data 8,8,8,24,8,8,8,0
1090 data 8,8,8,8,8,8,16,0
1100 data 72,72,72,240,72,72,72,0
1110 data 64,64,64,192,64,64,120,0
1120 data 254,84,84,84,68,68,68,0
1130 data 88,68,68,68,68,68,68,0
1140 data 126,66,66,66,66,66,126,0
1150 data 92,68,68,252,64,64,64,0
1160 data 126,66,66,67,66,66,126,0
1170 data 94,66,66,252,72,72,72,0
1180 data 126,64,64,62,2,2,124,0
1190 data 126,16,16,16,16,16,96,0
1200 data 108,34,34,34,34,34,62,0
1210 data 68,68,36,36,20,20,12,0
1220 data 68,68,68,84,84,84,254,0
1230 data 68,68,68,72,60,68,68,0
1240 data 72,72,72,240,32,32,32,0
1250 data 252,4,8,252,16,32,126,0
```

[NOTE] *The processor is little-endian so the byte that is stored first in the locations designated as address is the least significant and the byte that comes next is the most significant. Although visually to us humans it looks like the bytes are in reverse order, in reality it makes sense that the least significant byte is at a lower address and the most significant byte is at a higher address. In big-endian processors the opposite is true.* **Source: Simone Bevilacqua, "Le basi della programmazione"**

Bibliography:

Commodore 128 Oltre il Manuale, Rita Bonelli, Gruppo Editoriale Jackson Giugno 1986

C128 Programmer's reference guide, Commodore Business Machines, Inc., Bantam Book February 1986

A Christmas Tale

by Alessandro Gatti (DumpClub 64)

Why do we still insist on talking, discussing or even pontificating about old computers or video games? I don't think it's correct to reduce everything to a dive in the past or to the mere nostalgia of the old times. In fact, the discourse would all be exhausted in the short term, as happens with summer songs replaced each time by those of the next summer. There is, in my opinion, something else. Something more elusive and that better clarifies this motion of the soul of all retrocomputing enthusiasts: consolation! An emotion that satisfies a need that has always been primary on earth for all living beings, even if it is not officially listed among Human Rights (the omission is significant). Like enjoying Chaplin's or Laurel & Hardy's late night comedies, like a cold glass of beer after working hard in the sun. They bring relief, soothe fatigue, afflictions, or simply give us joy and pleasure, because in the end it is not the "juggling" of the graphics and sound of these old machines that enrapture us but the magic that they are able to evoke in waking us up from our daily lives, in reminding us of our imagination and our amazement. They soothe the disenchantment of adulthood and reassure us that we are still capable of astonishing and deluding ourselves, still able to dream, accompanying us on fantastic adventures!

As we say here at DumpClub64 these emotions are more contagious than any Omicron variant and above all they do not make us sick but they heal us from a reality that too often leaves much to be desired and find, precisely, what I was talking about: consolation. Christmas is the period in which we need a gargantuan amount of consolation, especially in these last years of tribulation, also because nowadays this feast is reduced to a blatantly consumerist event that clashes with the authentic values of spirituality that it should celebrate. Throughout the year we tend to get used to the noise of the sad reality that surrounds us, even if we unconsciously suffer from it, and around Christmas and at the end of the year the knots come to the comb and so, even for the most hardened hearts of life, this becomes a difficult occasion: what should have been a moment of joy actually becomes the day in which you are handed the bill, which is often a hefty bill!

It reminds us of those who are absent, and how much we miss them, how senseless are the worries that you chase throughout the year and reminds you how much time you have devoted to many useless things and how little you have reserved for what mattered most! Let me tell you that at such a juncture we need something that can cheer us up, and if it happens at Christmas time then we have all the ingredients for a beautiful Christmas story to put aside for when the winters of life will be more rigid. So to respect the tradition I would say to start it with the famous incipit that so many generations of dreamers have enchanted...

[...] Once upon a time there was an ice-house in the remote northern outskirts of Rome. In reality it should have been a hobby room, but the energy crisis and the consequent increase in electricity prices had decreed a microclimate similar to the Siberian steppe during the first ice ages. Among huge piles of junk and rickety stacks of cardboard





Fig. 1 e 2 - Orizzonti, educational software distributed by C.T.O. Amiga





Fig. 3 e 4 - Orizzonti, educational software distributed by C.T.O. Amiga

boxes, two cheerful figures were tinkering as usual with their favorite pastime: the Commodore 64!

Alessandro, completely wrapped up and with a blanket on his legs, was in the dumping station! The old PC, as big and heavy as a washing machine, had been picked up by Roberto in front of one of the many dumps that spontaneously arise in the capital. Alessandro, after cleaning it up and fixing it, had set it up to perform the only task that could give it a meaning: to rip from oblivion many C64 floppy disks. It was connected to a self-made Xumfloppy that in turn interfaced with a beautiful Disk Drive 1571, very rare according to Roberto, but that actually did its job very well!

"These discs are filthy but where did Francesco Tosti find them and where the hell are the cotton buds I brought you?" suddenly exclaimed Alessandro.
"The girls took them..." replied Roberto, from the second workstation dedicated to cassette dumping and equipped with all the comforts and conveniences of the case: C64 cookie with ceramic chip and 1541 Ultimate II plus, connected to a very rare 1702 monitor and a sparkling Cute32, in practice the cadillac of the workstations of all the retrocomputing fans.
"Alright! Then go out and go to the Chinese and buy some more," replied Alexander without fuss.
The idea of any task that could go beyond the usual copy and paste was for Roberto the equivalent of a sentence to the gas chamber! Launchotism (the syndrome of which Roberto was a healthy carrier all his life) did not allow him to roll up his sleeves and pushed him inexorably towards the lowest, most despicable and despicable forms of idleness that man had ever conceived.
"Maybe there's one left inside that little box..."
"You're the scum of the scum! You know that, right?"
"The world also needs pitching... I mean pitching also plays an important role: on so many occasions you also need that little bit of cross shield to solve..."
Suddenly the buzzer of a new message overbearingly draws the attention of the two.





Fig. 5 e 6 - Orizzonti, educational software distributed by C.T.O. Amiga





Fig. 7 e 8 - Orizzonti, educational software distributed by C.T.O. Amiga

"A message came in from a guy asking for our help in dumping an Amiga floppy disk," said Roberto. He also left us his cell phone. What should I text him?"
"I'm sorry he left your cell phone because you don't call him!"
"Ok but then you talk to him... if he asks me something too technical I answer him... if you are the expert on Amiga dumping..."
"In the meantime call him then in case put him on the phone..."

This is where the real fairy tale of this Christmas 2021 actually begins!

After calling that cell phone number, in fact, we met Nazario, a boy from Turin who, after reactivating his Commodore Amiga, was looking for a game that his father had bought him when he was a child. A game that had never been dumped probably because it was little known and of which the entire Amiga community did not even remember the existence! Unfortunately, the original diskette did not work, but thanks to the advice of his father, Nazario at the time had provided to make a backup copy! A valuable advice, like all the advice of a father to his son and that the same Nazario told us tries in turn to transmit to his children!

After explaining to Nazario the various procedures to dump the diskette we agreed together that perhaps it would have been easier if he had sent it to us. So after some difficulties and some delays of the courier we managed to receive his diskette and, as expected of a good Christmas story, we managed to dumparlo and make it available to all!

As Nazario told me, probably for those who have never seen it could seem an insignificant thing but for those who like him have seen it with the eyes of innocence during their childhood is something important: a warm blanket to cover and warm the spirit! Have fun with "Horizons" educational software distributed by C.T.O. Amiga!

Merry Christmas from DumpClub64 and Nazario!!!

The entire editorial staff of RetroMagazine World supports and appreciates the work of the guys at DumpClub64.

We invite our readers to collaborate with them in case they have material to preserve and/or share with all retrocomputing enthusiasts.

BASIC fractals in 8-bit sauce - Part 2

by David La Monaca

Let's continue our little journey among fractal images and graphical figures obtained using the several BASIC dialects of the most popular and famous microcomputers of the 80's and 90's. Some of our most attentive readers have responded to our call to expand the list of 8-bit listings able to reproduce the "classic" fractals and other pleasant and particular 2D high-resolution figures. Among these, we would like to point out some listings for Commodore 8-bit machines sent by **Armando Pavese**.
In fact, the VIC-20 and C-64 were somewhat snubbed in the first article of this column, mainly because, as is well known, the standard BASIC V2 does not include graphical commands for high resolution. Not that it is impossible to plot curves and drawings, including fractals, using these systems through direct interaction with the memory locations of the graphical page, but certainly the listings and the related procedures for activating high resolution, drawing lines and points, setting colors, etc. would have been quite complex and unreadable.
Armando, in the listings that he kindly sent us, exceeded the limit of the BASIC V2 of the C-64 using Simons' Basic, the famous extension provided by Commodore itself, while for the VIC-20 he used the graphic commands and the additional functions offered by the Super Expander cartridge (in the version sold in Japan, marked by the label VIC-1211M, that provided to the unexpanded VIC-20 also 3KB of memory expansion). Armando told us that he preferred the syntax of the instructions of this version compared to the European version that was marked VIC-1211A and that actually featured a rather different set of commands.

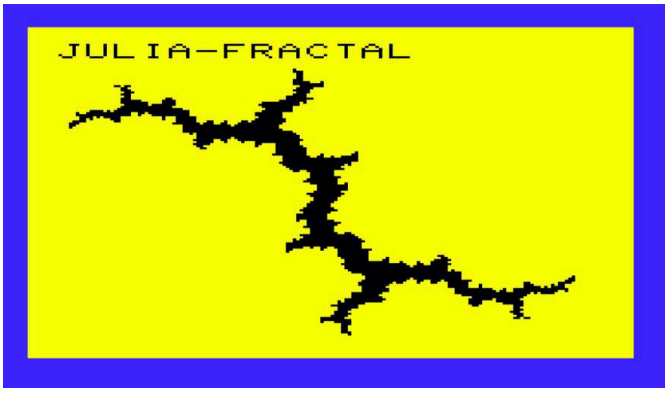
Among the other listings published in this issue, we point out the Mandelbrot Set traced from a listing for C-64 + Simons' Basic by our **Marco Pistorio**, who had dealt with it in issue 03-IT (RetroMagazine Italia published in January 2018). We republish it here corrected and tested (at the time it ended up on the pages of the magazine with an inaccurate final part of the listing).

So without putting further time in between, after thanking Armando and all those who sent in their work, here is a second selection of fractals and geometric figures for your and our beloved 8-bits!

ERRATA CORRIGE - In the last issue, the listing n. 05 "Barnsley fern" for MSX2 was published with a small error that affects the functionality of the program. We propose here below the correct BASIC line to replace the published listing.
Correction - line 80:
80 T=-.15*X+.28*Y:Y=.26*X+.24*Y+.44:X=T:C=3

06. VIC-20 + SUPER EXPANDER JP VIC-1211M (6502, 1981) - Julia set
(porting by Armando Pavese)

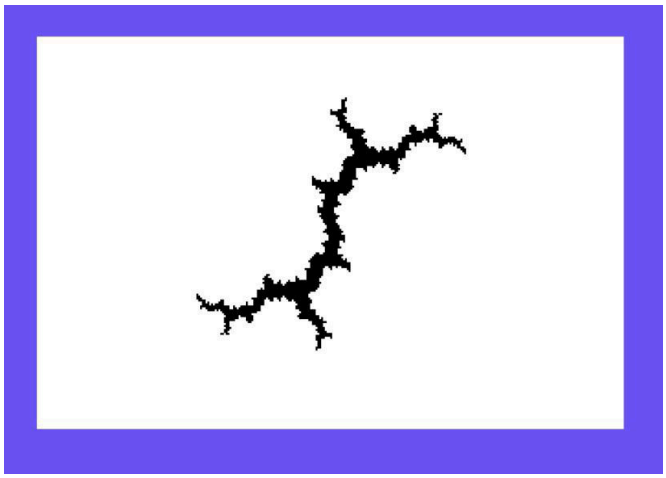
```
10 hires1
20 fori=-75to75
30 forj=-75to75
35 setc1,8,7
40 ploti+85,j+75
50 zr=i/56:zi=j/56
60 k=0
70 te=zr*zr-zi*zi
80 zi=2*zr*zi+1
90 zr=te:k=k+1
100 ifabs(zr)<2andk<9then70
110 ifk<9then:setc8
115 ploti+85,j+75
120 nextj:nexti
130 setc1
150 char 1,1,"julia-fractal"
200 wait198,1
210 text:end
```



07. C64 + SIMONS' BASIC (6510, 1983) - Julia set
(porting by Armando Pavese)

```
10 HIRES0,1
20 FORI=-75TO75
30 FORJ=-75TO75
40 PLOTI+160,J+95,1
50 ZR=I/56:ZI=J/56
60 K=0
```

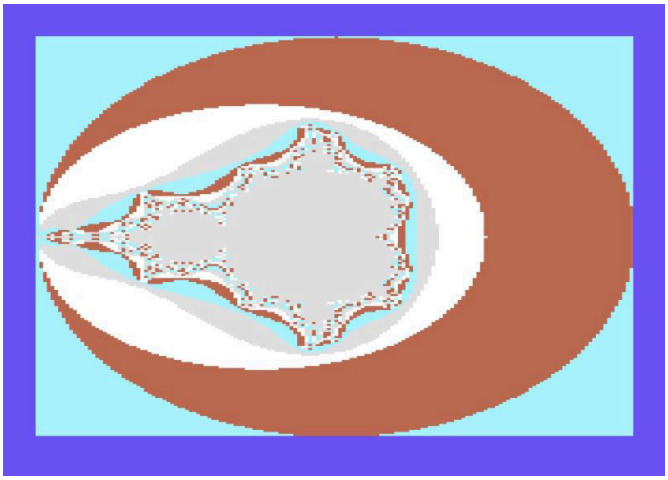
```
70 TE=ZR*ZR-ZI*ZI
80 ZI=2*ZR*ZI+1
90 ZR=TE:K=K+1
100 IFABS(ZR)<2ANDK<9THEN70
110 IFK<9THEN:PLOT I+160,J+95,0
120 NEXTJ:NEXTI
200 WAIT198,1
210 NRM:END
```



08. C64 + SIMONS' BASIC - Mandelbrot set

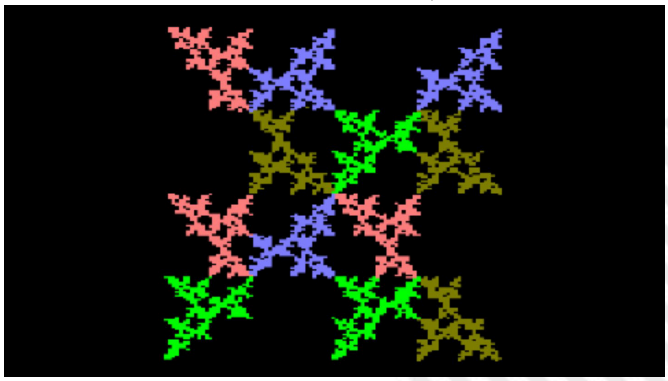
(by Marco Pistorio - RM 03-IT)

```
10 IX=-2:SX=2:REM COMPLEX AREA OF THE PLANE
20 IY=-2:SY=2:REM TO BE DISPLAYED
30 MC=32:REM NUMBER OF ITERATIONS
100  HIRES  1,0:MULTI  3,2,1:  REM  ENABLES
MULTICOLOR GRAPHIC SCREEN
110 :
120 LX=SX-IX: LY=SY-IY:REM RANGE OF X AND Y
130 XX=LX/160:REM X RANGE RESOLUTION RATIO
140 YY=LY/200:REM Y RANGE RESOLUTION RATIO
150 :
160 FOR I=0 TO 160:REM LOOP FOR EVERY X VALUE
170 CR=IX+(I*XX):REM COMPUTES REAL PART OF C
180 FOR J=0 TO 200:REM LOOP FOR EVERY Y VALUE
190 CT=0:REM COUNTER
200 ZI=0:REM IMAGINARY PART OF Z
210 ZR=0:REM REAL PART OF Z
220 CI=IY+J*YY:REM IMAGINARY PART OF C
230 :
239 REM APPLIES MANDELBROT FORMULA
240 IF((ZR*ZR+ZI*ZI <=4) AND (CT<MC)) THEN
GOTO 300
250 CL=CT:PLOT I,J,CL AND 2+1:REM PLOTS DOTS
OF COORDINATE I,J WITH COLOR CL AND 3
260 NEXT J
270 NEXT I
280 GETA$:IFA$=""THEN 280
290 END
300 :
310 TZ=ZR*ZR-ZI*ZI+CR:REM NEW REAL PART OF Z
320 ZI=2*ZI*ZR+CI:REM NEW IMAGINARY PART OF Z
330 ZR=TZ:CT=CT+1:REM INCREASES COUNTER AND
GOES ON
340 GOTO 240
```



09. AMSTRAD CPC LOCOMOTIVE BASIC - CHAOS GAME

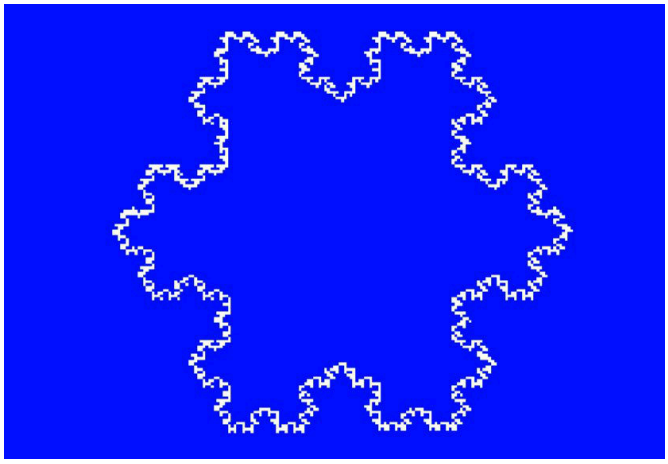
```
10 MODE 0:INK 0,0:BORDER 0
20 x=360*RND:y=360*RND:last=2
30 FOR i=1 TO 15000
40 vertex=1+INT(4*RND)
50 IF vertex=last-1 THEN 40
60 x=x/2:y=y/2
70 IF vertex=2 OR vertex=3 THEN x=x+180
80 IF vertex=3 OR vertex=4 THEN y=y+180
90 PLOT 139+x,19+y,7+last
100 IF vertex=1 THEN last=5 ELSE last=vertex
110 NEXT
120 CLEAR INPUT:WHILE INKEY$="":WEND
```



10. ORIC-1 - KOCH SNOWFLAKE

```
10 HIRES:PAPER 4
20 DIM S(20)
30 A=0:D=PI/3:X=84:Y=164
40 FOR I=1 TO 3
50 A=A+2*D
```

```
60 S(0)=140:SP=1
70 GOSUB 100
80 NEXT
90 END
100 SP=SP-1:L=S(SP)
110 CURSET X,Y,1
120 IF L<=5 THEN H=L*SIN(A):V=L*COS(A):DRAW
H,V,1:X=X+H:Y=Y+V:RETURN
130 L=L/3
140 FOR P=SP TO SP+3:S(P)=L:NEXT
150 SP=SP+4
160 GOSUB 100:A=A-D:GOSUB 100:A=A+2*D:GOSUB
100:A=A-D:GOSUB 100
170 RETURN
```



11. ATARI TURBOBASIC - Lorenz Attractor (fig. 1)

```
10 GRAPHICS 8+16:SETCOLOR 1,1,14:COLOR 1
20 R=10:S=28:B=8/3:D=2.0E-03
30 X=0.1:Y=0:Z=0
40 FOR I=1 TO 16000
50 XP=160+5.7*X:YP=96-3*Y
60 IF XP>0 AND XP<319 AND YP>0 AND YP<192
70 PLOT XP,YP
80 ENDIF
90 XX=X+D*R*(Y-X)
100 YY=Y+D*(X*(S-Z)-Y)
110 Z=Z+D*(X*Y-B*Z)
120 X=XX:Y=YY
130 NEXT I
150 IF PEEK($D01F)<>6 THEN 150
```

The second part of this column ends here, but we still have many listings in store for you. We still await your ideas for fractals and geometric designs of interest to 8-bit microcomputers.

Have you experimented with anything similar? Send us your listings to the email box:
RetroMagazine.Redazione@gmail.com
All works will be considered and the best ones will be published in the next issues.

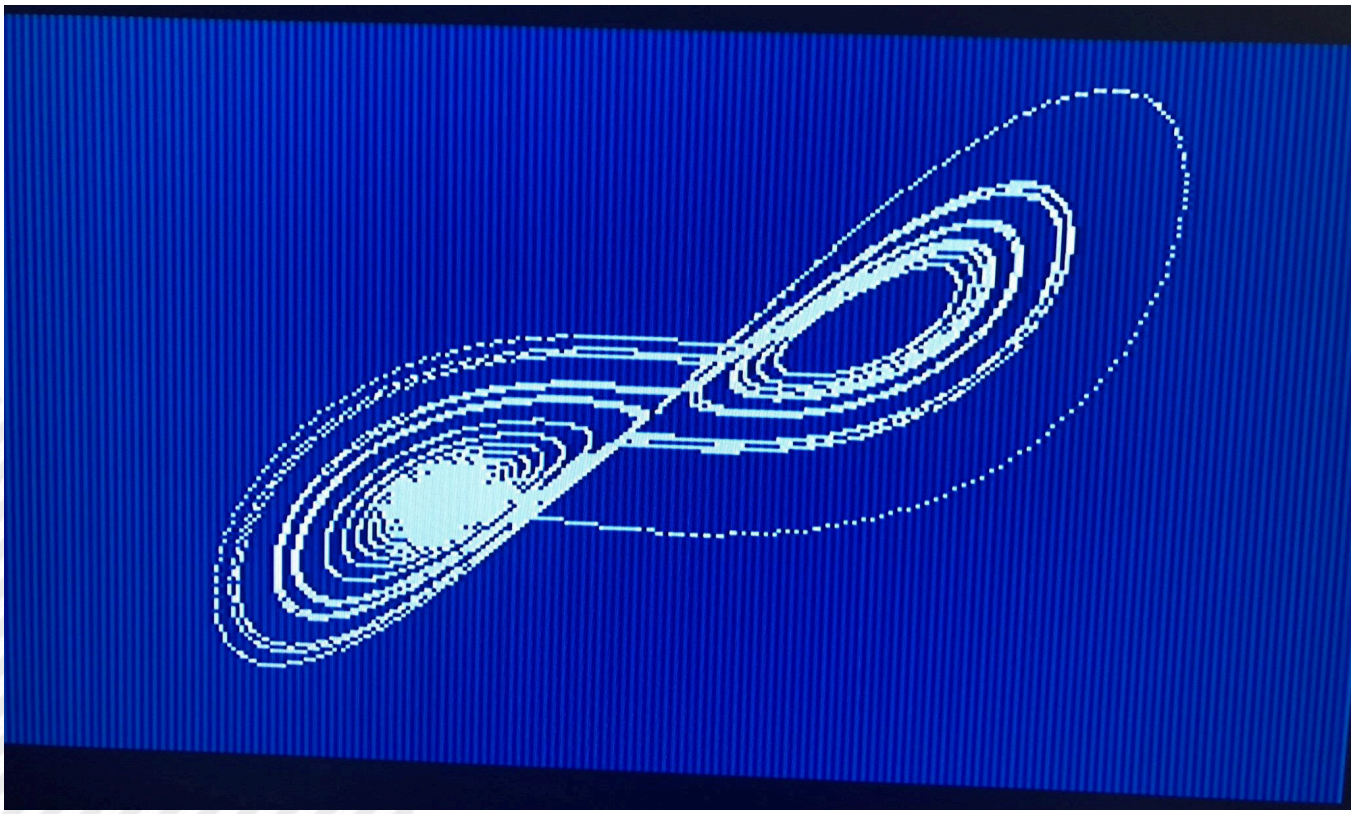


Fig. 1 - Atari TurboBAsic - Lorenz Attractor

Commodore 64's guidebooks re-edition

by Carlo N. Del Mar Pirazzini

Despite the fact that retrocomputing on the Internet has become in the last 10 years a huge receptacle for wackos, conspiracists and gossip lovers, every now and then it still reserves us something really good.

That's the case of a page that I link here at the end of this mini-review. It is the web page of a blogger with a passion for the world of retrocomputers and in particular the Commodore 64.

A passion so great to have made him embark on the titanic enterprise - which by the way was completed - to rewrite and reprint two very interesting guides for our beloved "breadbin" home microcomputer. I'm talking about the new versions of the **C-64 User's Guide** and the **Programmer's Reference Guide**. So far, of course only the English versions are freely available for download.

He could have simply scanned the old manuals and put them on the Web, but he bravely decided to make something from scratch starting with his copies as the main source. He completely rebuilt (including covers), every log map, every diagram, every table and every paragraph achieving a flawless end result.

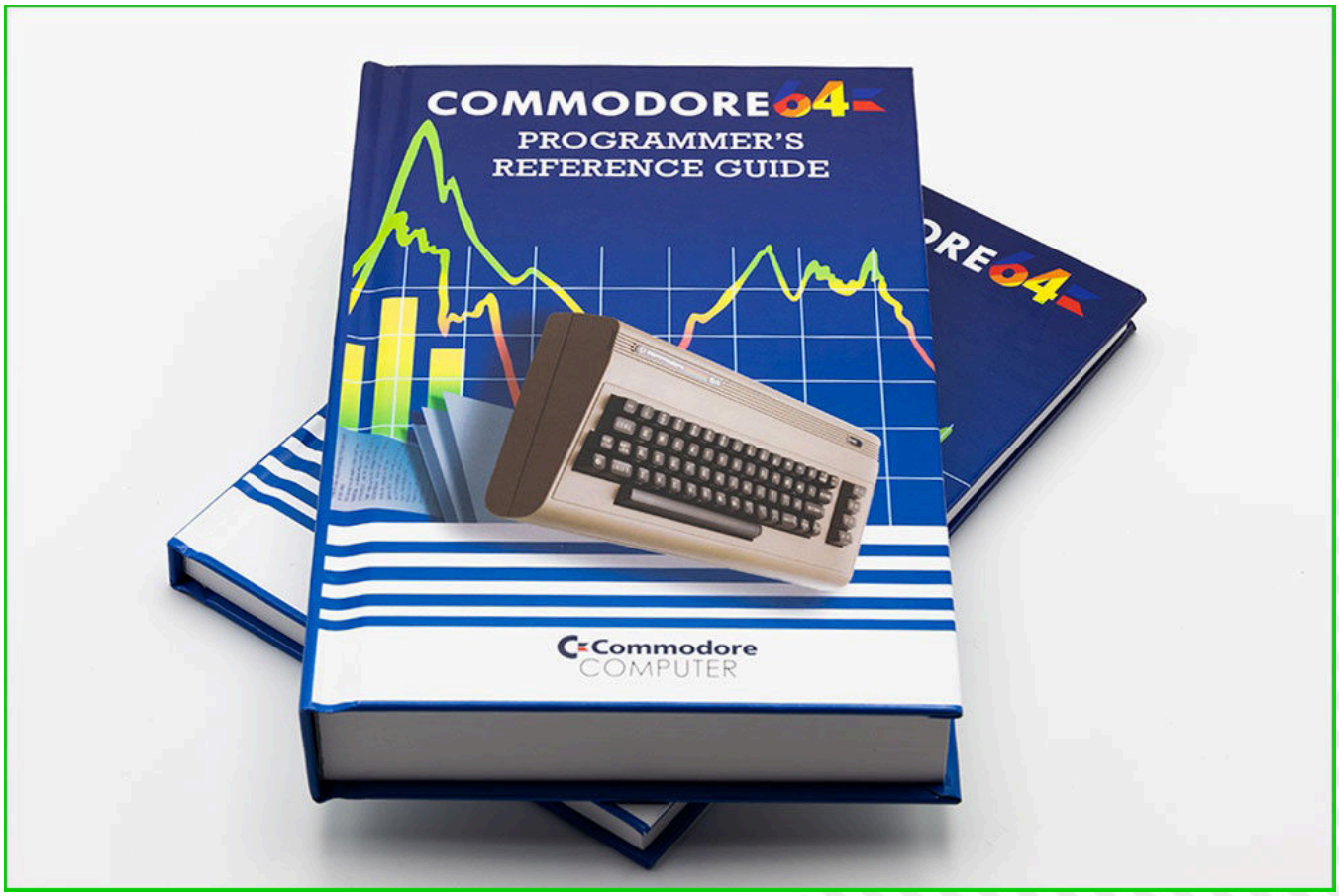
A work that probably took months and that is still carried on with continuous corrections and additions.

He even tried to print a couple of copies by himself, but due to some problems with the typographical output, he couldn't.

Anyway, it generated two wonderful print-ready PDF files complete with covers. The author has already successfully experimented with lulu.com as a printing service, but other Print On Demand sites should be fine too.
What can I say, this is an incredible job, to say the least!

You can reach the Pickled Light blog site at:
https://pickledlight.blogspot.com/p/commodore-64-guides.html

The updates and corrections keep being published so be sure to visit the blog from time to time.

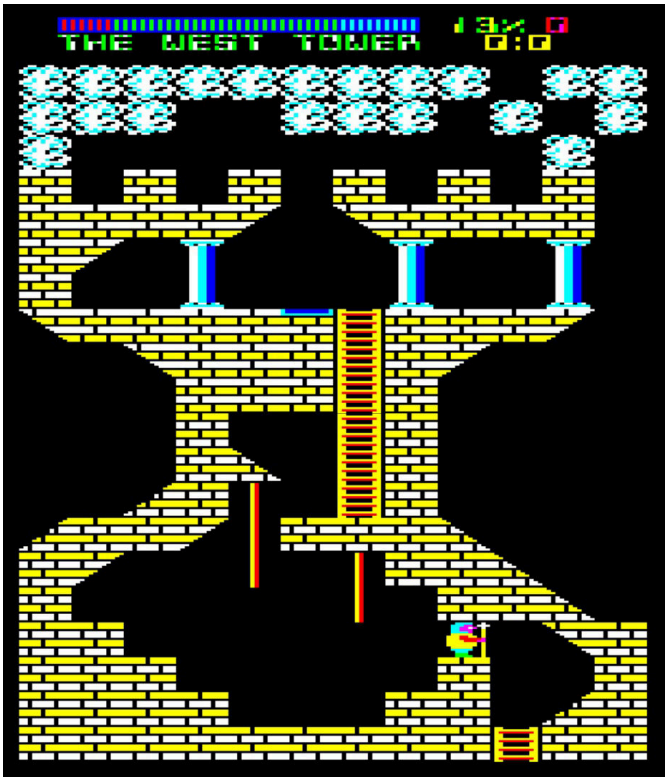


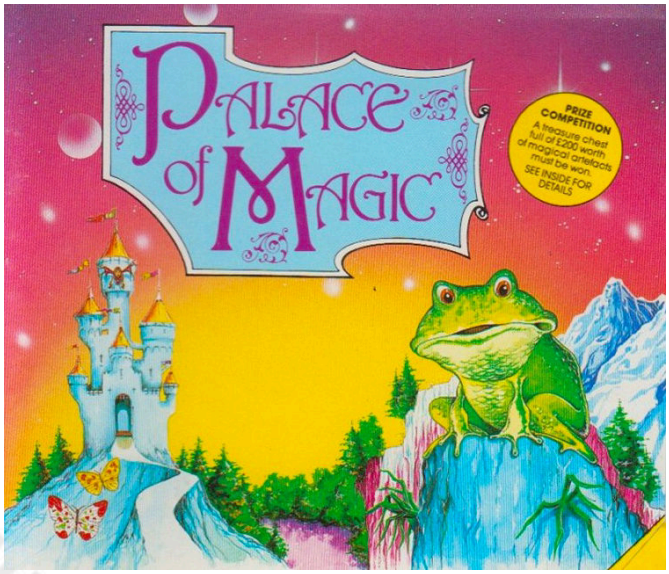
Welcome to the Palace of Magic
(and the story of the 8-bit graphics card)

by Robin Jubber

I collect so many elderly computers and dust covered consoles that I have ended up with several piled on top of each other, all waiting for their day in the sun. Unlike a lot of other collectors I don't collect much of the software, preferring SD card solutions that allow machines like the Dreamcast, PC Engine and CD32 to boot up with all their precious titles on display. That also appeals to my inner electronics wizard, despite my incompetence with a soldering iron. Would anybody like to buy a very recently broken Amstrad GX4000? I have two... However when it comes to boxed games there is one machine for which I make an exception. My first true love, the BBC Micro. Even then there are just a few Beeb games that I have space in the Lab to put on their own shelf, and the most precious of these is a game you have probably never heard of, called Palace of Magic. Famous or otherwise, it's the perfect lens for a deep-dive into 8-bit nostalgia.





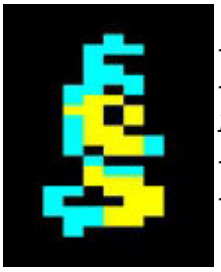




When I was thirteen and probably on less than a pound a week pocket money, it took me months to save up for this game. But I simply had to have it, no matter the pecuniary obstacles, because the art drew me in completely. Look at that frog! Look at that castle! You can almost reach out and touch the palace in the distance, set against a gorgeous evening sky. I could probably draw that image in my sleep, 15 crescent moons and all. Box art was a crucial part of the 8-bit experience that slowly began to lose importance as the 16-bit era rolled on. It was how you prepped your brain for the adventure to come. For 8-bit machines, the gulf between the art and the game

itself was huge, with the player's imagination acting as an 8-bit graphics card that filled in all the blanks.

These days games are often released without a cover image or box at all. This all feels like a step backwards. Even when they are accompanied by box art, it simply isn't having to work the same magic. The graphics in the game will look as good as any mocked-up screenshot or hand-drawn image. When I look at the game screens for PoM over three decades later, there's still a visceral thrill at these evocative locations such as The Gardens, The Rooftops and The Kitchen, that isn't solely down to the actual art on display. To a less nostalgic eye, the graphics are simplistic; repeated textures held together with maybe a palette swap or some subtle atmospheric details like a glowing hot oven defined by a two sparkling dots. To the young games player of the 1980s however these were real places and utterly bewitching. The heart of exploration with early arcade adventures was a reward structure where simply finding a new location did the same job that a glowing epic sword has to carry off in World of Warcraft. Maybe, like with the titans of the genre (Legend of Antiriad, Exile, Jet Set Willy) there was even the hint of a plot. I think I must have marvelled at the Exile novella's gory artwork and re-read the contents tens, if not hundreds of times. In Exile I knew everything about Triax and his cobbled together robots – with each encounter I was re-reading the relevant chapter in my head. Die you brutes - you won't take me alive! Tiny little 6 by 20 sprites in perhaps four colours still have the capacity to send shivers up my spine. Enlarge the main character in one of these games and it's a real feat of imagination to see what's going on. This is where the game's creators were forced to co-opt 86 billion organic circuits and millions of years of evolved pattern recognition to fill in the gaps. These days we no longer need our fleshy GPU to work nearly so hard.



- What's that daddy?
- It's a man in a spacesuit! Obviously. Isn't it great?
- It looks like crap daddy!
- Go to your room!

Another wonderful aspect of 8-bit games was that they were often the first of their kind. It was not at all unusual to discover games called Adventure, Maze, Pilot, Spaceship or Racer, especially on the less well supported platforms. If your parents were the sort of well-meaning sociopaths who bought you an Einstein or an Oric, the only game you could buy for the system was probably called 'Game'. The company who published it was probably called Oricsoft. The necessity for a complex hyphenated name like "The Elder Scrolls V: Skyrim" or "Raid: Shadow Legends" was years away. There was room to manoeuvre. Now the only remaining game with a title like an 8-bit classic is perhaps Flight Simulator, from Microsoft. That's a good stolid down-to-earth name you can trust. With my beloved Palace of Magic there's no confusion. You don't need to be familiar with a multi-title sequence or a publisher's output to understand what's going on. This is a game set in a palace. With magic.

Bring a sword

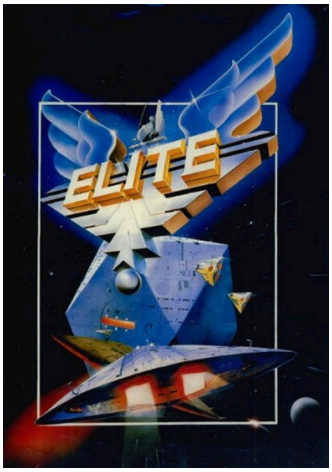
This means of course, unless some forgotten tribe is just now discovering abandoned 8-bit machines that have inexplicably floated past their village, the era of the 8-bit graphics card is gone forever. We can never return there. 8-bit graphics were a mixture of whatever the machine could offer, which by today's standards is next to nothing, and everything your fevered imagination could bring to the table. The simplest games my 3-year-old has played, Velocity and Tomb Raider on the PSP or 2D Switch titles such as Iconoclasts and Owlboy, are still light years ahead in terms of visual sophistication. There's almost no interpretative layer required. As for the free software on her iPad – forget about it. Fully 3D rendered Unity gems, every single one of them. With in-game advertising. How far we've come.

Whilst pixel art games do get made today, they usually have a larger palette than the original 8-bits – plus they're not the cutting edge, but more a charming throwback that sits in an ecosystem of advanced gaming. You're not seeing something on the edge of the possible. We know there are visually better games out there. Crysis can't be un-invented.

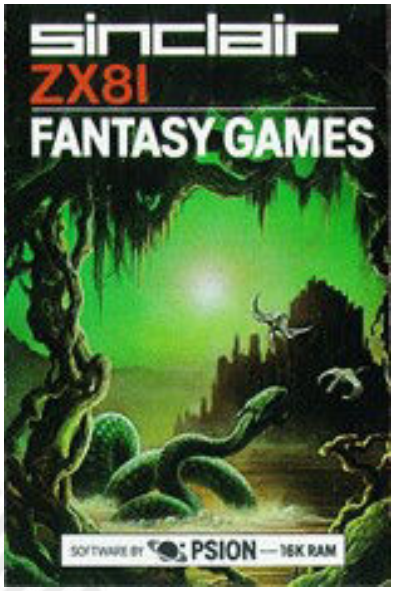
One of the key components to grabbing the attention of a 70s or 80s gamer was the poster or box cover that







accompanied these single programmer masterpieces – the trigger for your imagination to fill in all the details a Motorola 6845 or MOS VIC-II couldn't handle. Many games of the era managed this trick wonderfully. My personal favourites were the cover art for Elite, Codename Droid, Atic Atac and the haunting sci-fi visuals of Alien 8. But I'm willing to be educated outside my original machines and I'm hoping the staff of Retrogaming Magazine will add their favourites to the bottom of this article. The Last Ninja box art was great on every platform and perhaps the greatest misrepresentation ever has to be "Fantasy Games" on the ZX81. Never has a box cover lied so artfully about the blocky horrors within.



<-- Lies

My daughter will one day have the fine motor control to play Palace of Magic on the BBC Master she currently eats her dinner off, but I'm doubtful she will be drawn into that glorious 8-bit castle in the same way I was. However she does have the box art in a massive poster on her wall, so that at least is something we can share. The colours are as vibrant as ever, the distant castle still enticing. Nothing triggers the nostalgia that makes this such an enticing hobby as the bold and beautiful box art of our youth. Even now that picture still has the power to make me yearn to revisit the Palace of Magic.

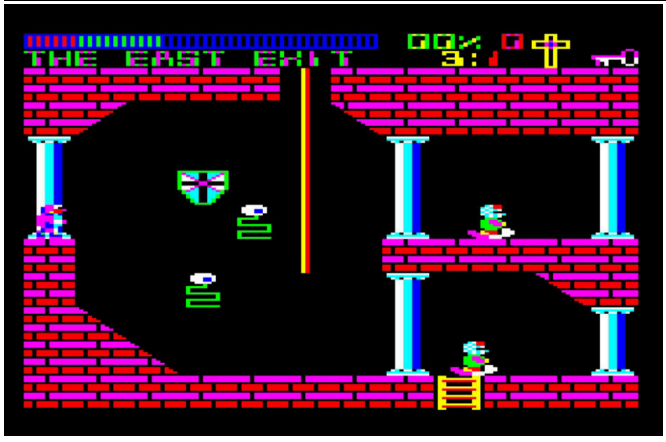
Robin's plea has not gone unheeded by the RMW staff, so here are our preferences for favourite and forgettable box art!

***David** - Best: Raid Over Moscow (C64), Out Run (ZX Spectrum), Starflight (Atari) - Worst: Phoenix, E.T. (2600) - I also like to mention the cover of Impossible Mission (C64), which, following a reverse logic featured a box art not up to the game.*
***Nithaiah** - Best: Elite, Uridium, Raid Over Moscow*
***Francesco** - Best: Turrican, Denaris, The Last Ninja - Worst: Park Patrol, The Great Giana Sisters, Tetris*
***Michele** - Best: Hawkeye, Armalyte, International Karate +, Antiriad*

*Send us your podium of the best (or worst) titles in the ratio of cover graphics to on-screen graphics (**retromagazine.redazione@gmail.com**).*







Japan 18th issue: Nintendo against all

by Michele Ugolini

Dear readers, are you ready for a spicy episode? My absence in these last few issues of RMW has given me the idea to narrate, through the Japanese philosophy, a delicate topic of Nintendo. I will tell everything through their marketing logic, focusing no longer on the laudable aspects of loyalty, census, recruitment and addiction.
I will analyze these logics highlighting above all the aspects of dependence, deviation from the standard, the identification of our friends from our enemies, the drama of having to put up with our enemies who are scarring the content created by our friends, the sharing of the punitive decisions made against the enemies, the obtaining of the most total justification with lots of "roaring applause" at the oversized epilogue of the eventual punishments against our enemies, which came along the way.
Friends and enemies, in what sense?
We all remember that Nintendo, in recent years, has actively produced a punitive movement against numerous illegal servers that hosted the pirated ROMs of the big "N".
Let's play a collective game. Let's begin to direct ourselves neatly, as a good consumer flock, into the precincts of "friends" and "enemies" to decide to criticize or applaud the decisions made by Nintendo.
We take our time deciding which side we are on, but we leave the fences open, because in some chapters of this document we might want to get out of one of the two fences.
So, we were talking about an illegal server and the actions against it.
There was talk of a punitive movement. "Rightfully" punitive, some of you readers are thinking.
Obviously we're all friends and we're punishing common enemies: the odious enemies of the big "N".
Do you all think that? Or do only some of you actually think that?
Are we really in the right position to objectively observe the ranks of supposed enemies, true enemies, those who command friends, and especially true friends?
Life alas is not a video game, pirated ROMs were obviously a reality and the punishments were just as real.
In this case, however, the amount of punishment was quantified through an obscure purpose and was directed to the sole advertising goal of "punishing the few perpetrators with harsh scorn, with thunderous and extensive news coverage, across the most massive number of social platforms, to educate all remaining dissenters."

:: WE INVADE LOVERETRO AND LOVEROMS?

Let's catch up on the historical basis for all of this: in nearby July 2018, the Big N was preparing to take legal action against two major, more or less legal ROM distributors. The reason was very simple. A huge infringement of rights, or in other words, a huge amount of money not paid into the coffers of



Fig. 1 - Nintendo against all

big "N".
The sites in question are LoveROMS.com and LoveRETRO.com. We are all very familiar with these sites related to the video game ROM circle.
The answers came quickly: LoveROMS was ready to omit all Nintendo products from its showcase, and LoveRETRO was just as ready to close the situation by sending everything offline with this message to all visitors:
"LoveRETRO is closed for all purposes until further notice. Thank you for your support, and we hope this will be resolved."
"LoveRETRO has effectively been shut down until further notice. Thanks for your patronage to date and we hope to get this figured out."
The lawsuit was filed against the two websites in an Arizona federal court. The complaint states:
"The LoveROMs and LoveRETRO sites are among the most open and notorious crossroads for pirated games. Through the LoveROMs and LoveRETRO sites, the defendants reproduce, distribute, publicly present, and display a shocking number of unauthorized copies of Nintendo video games, all without Nintendo's permission."
Here is the link to the news story:

https://torrentfreak.com/nintendo-sues-console-rom-sites-for-mass-copyright-infringement-180720/

"The LoveROMs and LoveRETRO websites are among the most open and notorious online hubs for pirated video games, Through the LoveROMs and LoveRETRO websites, Defendants reproduce, distribute, publicly perform and display a staggering number of unauthorized copies of Nintendo's video games, all without Nintendo's permission."
LoveROMs and LoveRETRO have been involved and justice has been done? This "shocking" number of copies has realmete created a "shocking loss of earnings" in the coffers of the big "N"? Or these ROMs were hosted to revive some nostalgic restored cabinets in the garages of us retrogamers? Why don't we talk about RomUniverse too?

:: INVADE ROMUNIVERSE?
Summer 2021 has just ended and Nintendo in July was again involved in similar victorious, or rather, tragicomic news announced by the following headline: "Nintendo won a $2 million lawsuit against a ROM site but can't even get paid $50 a month: The legal battle continues".
Suing a video game rom site and receiving $2.1 million in damages yields little revenue if the people involved in the lawsuit don't even have the money to pay their monthly living expenses, in fact Nintendo is having trouble receiving the $50 monthly from RomUniverse founder Matthew Storman.
In fact, poor Matthew has already missed the first payment of $50 in monthly penalties to the video game publisher.
Do you remember the start date of this lawsuit? It was in 2019. Well, earlier this year that Nintendo's lawsuit against the RomUniverse site finally ended with the site being shut down and fined. The calculation of the fine is very simple $35,000 for 49 Nintendo games on the site along with the aggravation of $400,000 in damages to big "N".
Matthew Storman obviously defended himself claiming that the income, obtained from that site, was so small that the amount calculated to pay the fine could only make... the entire court smile.
In fact, recently leaked documents of Nintendo complaining about a "very serious and unexpected bureaucratic hitch", it seems that "incredibly and incomprehensibly" Storman has not made the monthly payment of $ 50 established by the court. Unacceptable!
In the document, Nintendo seeks a permanent injunction against Storman to ensure that it does not relaunch RomUniverse and resume distributing video game roms.
We can view that document in its entirety on Torrenfreak, we can read (with tragicomic amusement) that Nintendo's lawyers cite Storman's inability to give them $50 as "evidence of his disregard for his legal obligations."
They make it clear, in their view, that "Not only has the defendant failed to address his violation of the Court's order in the opposition, but he continues to ignore the order and has yet to make any payment to Nintendo." Obviously, this viewpoint of bureaucratic humor is not at all consistent with the reality of a disproportionate fine levied against a real-world person. Therefore, to us readers, end users, outside the perimeter of this dispute, what news is reaching us? Simple: this non-payment, albeit a modest $50 per month (the amount proposed and agreed upon by the parties), demonstrates that Nintendo has no adequate legal remedy for defendant's past or future infringement and underscores the need for a permanent injunction.
Nintendo is famous for these disproportionate fines: usually these lawsuits end with millionaire verdicts in favor of the Kyoto company.
In reality, however, the people charged fall into a comical situation of insolvency. Could this situation deter big "N" from pursuing the legal battle?
Obviously not. Dear readers, what did the judge decide in the end? Simple: in the mathematical, bureaucratic and imaginative reality of Nintendo there are 2.1 million dollars of lost revenue, but in the reality of people physically living on this planet

Earth ... there are other ways. In fact, the judge (also terrestrial) has obviously decided not to issue any permanent injunction against Storman, which is why the portal was closed only for a limited period of time and then it is back in operation again, leaving Nintendo with a lot of sadness and huge doubts about "how it was possible not to have received 2.1 million dollars in their coffers.

Nintendo, with a purely Japanese mentality, just could not understand the reason for this bureaucratic hitch: it had won the case but had not been able to recover the lost profit nor to stop everything that the culprit had committed and especially what he will commit in the future.

A curious fact: Matthew was unemployed, and would never be able to pay the 2.1 million dollars owed, which is why the amount was divided into convenient monthly installments of 50 dollars (42,000 installments in total) to be paid in the future ... 3,500 years! Tragic situation, isn't it? By the way, how is the movement going in the precincts of Friends of Nintendo and Non-Friends of Nintendo?

:: INTEL INSIDE, NINTENDO INSIDE, INSIDE SEGA OR INSIDE ANDROID?

If you think Nintendo is an inferior reality to Sega, Playstation, Naomi, etc... you are totally wrong. Nintendo is a power to be reckoned with and its roots are happily and rapidly spreading in a ubiquitous direction. On all platforms, even on "rival" platforms. Also know clearly that Nintendo has no rivals. Big "N" lives on marketing and moves based solely on verifiable algorithms, continuously studying the possibility of earning "any" type of loyalty to its customers. This method is also a purely Japanese philosophy.

Maybe that's why Sega and Microsoft are shaking hands coldly in this abnormal winter?

A few months ago SEGA and Microsoft signed a strategic alliance that will allow the Japanese company to carry out the "Super Game" project, which was announced in May.

"We are proud to announce that we have laid the groundwork for a strategic alliance with Microsoft to implement the "Super Game" project and create a next-generation game development environment. Our goal will be to further advance the development of our titles so that they can be enjoyed by fans around the world. We therefore aim to build an alliance that utilizes both SEGA's powerful game-making capabilities and Microsoft's cutting-edge technology and development environment. (Yukio Sugino, President and COO, SEGA Corporation)."

We don't know much about it yet, but the agreement is a fundamental part of SEGA's medium and long term strategy in the creation of new and innovative games, focused on Azure (Redmond's proprietary cloud technology), whose key points will be "Global", "Online", "Community" and "IP Utilization". Microsoft for its part will actively collaborate in the project by providing the Azure cloud platform to continue its development and optimization issues. The two big companies will cooperate to build new network technologies and communication tools with a global approach. Big words that leave room for wide imagination.

Indirectly then, involving the "PC" platform, could represent for Nintendo a fantastic opportunity as at the time "Mario" for Android?

Remember "the new Mario game you can tackle with one hand?" that came out a while back? The app states that "Mario advances relentlessly through each level performing a variety of jumps. Depending on how you touch the screen, Mario will move differently, perform skillful moves, and collect coins, finally reaching the finish line!"

We know that Mario is now an undisputed eternal icon of big "N".

Now let's try to shift the focus to Zelda. By now, Link and Zelda are also two icons that have become eternal like Mario and Luigi, right?

Well, how many gamers have been fantasizing for years about the possible arrival of Nintendo games on PC? So everyone could enjoy some titles with superior graphic quality thanks to the latest generation hardware and achieve results similar to the latest Zelda Breath of the Wild video in 8K with Ray Tracing.

From the web, a recent leak, may suggest the arrival of a collaboration between the Big N and Nvidia to bring some of the most popular games to PC via the GeForce Now service.

This leak from Nvidia's company is just great! Will it be real?

In the leak you can find clues about New Super Mario Bros. Wii and Mario + Rabbids Kingdom Battle, maybe they could even be pioneers of numerous other collaborations and software developments, who knows!

Remember the leaks that argued the arrival of the PC version of God of War or the existence of Grand Theft Auto The Trilogy The Definitive Edition?

This new leak could be just as true. Obviously big "N" does not confirm nor deny: a classic method to say that in the house of Kyoto, everything is possible.

:: NINTENDO CONQUERED THE SALES TOPTEN BY CHANCE?

How many more achievements are nearing completion in the home of our beloved Nintendo?
The first news is from a few weeks ago: Nintendo Switch has surpassed PS2, becoming the fourth best-selling console ever in Japan with 22.03 million units placed since launch on March 3, 2017.
Last August, Nintendo Switch had sold about 21 million units in Japan,
incredibly then in a few months has sold over a million pieces, coming to exceed 21.98 million PS2
The second news is that the surpassing of 22 million Nintendo Switch was realized in 246 weeks. We can remember that in first place in the stopwatch record momentarily there is only NDS with 170 weeks.

:: DO WE KEEP PRICES SKY HIGH?

Remember the macabre and censored Mortal Kombat?
Why are the new MKs less on the market than our beloved , old, fun and colorful Mario?
Simple: Nintendo has managed to take advantage of the ups and downs experienced over the past 25-30 years to
Maniacally perfecting the fruit of their work, playing on personalities and branded products that can generate an automatic memory about the identity of a name pronounced aloud: "Mario!". Very evocative! A discourse of exclusivity, branding and loyalty. On the other hand, it is just as evocative to talk about Mario Kart racing, Super Mario platformers and Party Games.
In the last few decades, big "N" has become a reference point for all other video game companies: who among us could list an unforgettable PS or XBOX saga that at the last chapter ended with excellent reviews? Probably all of them ended with a thud into the abyss, consumed by their own soul, spun and stretched so many times that in the end the fatal irreparable tear occurred.
Who has really reinvented himself in his success, always using the same characters? We can count an eternal hero in Pac Man, in Sonic, in Mario and a few others. We can then come to a calm inner reflection about the constant search for the "new phenomenon": the constant pursuit of the new trend and that sometimes obtuse desire to evolve the brand has seen consume and forget so many sagas, apparently more appetizing, but in the end became more fickle in their matrix.
Big "N" knows very well that during the gift-giving season, a hypothetical uncle will gladly buy a classic "Mario" for the family: so the grandchildren can enjoy colorful and healthy fun without blood or violence. Nintendo, of course, has always produced what an in-depth marketing study deemed indispensable for the customer.
Big "N" knew very well that if its customers have certain purchase needs (colorful, fun, socializing games, etc. ..), why would these loyal customers pay less for a video game from the previous year, perhaps sold out from the shelves? If a game is no longer for sale, Nintendo might consider it a "collector's item", so should the price decrease or, more likely, should it increase? Nintendo doesn't devalue its games because it doesn't inflate its franchises, if anything it is the collectors' public that feeds the increase in price of past games, in the murky environment of collecting.

Moreover, reflecting calmly, today, after almost 5 years, we are still waiting for the sequel to Zelda that should be out in a few months: why then Nintendo should feel obliged to lower the price of a product that took so many years of development, which has no equal on the market and is still the last main episode of its saga?
Nintendo is good at fueling anticipation, cultivating the brand, and making each release eternal.
Big "N" doesn't allow any video game to be released without a well-established marketing plan: every release must be "perfectly calibrated."
It's not enough that the game is running on stable software to be truly ready for distribution: the game must undergo numerous variables determined by big "N" before it sees the light of day on the shelves.
When in the other houses the birth of a new game becomes a methodical process, repeated in the short term, predictable, then it happens that the value of the brand is diminished in the eyes of the consumer: this rule of marketing is considered "sacred" at Nintendo.

Another murky aspect of marketing at home big "N" we find it in the management of the prices of the Nintendo Switch Online service, all in all accessible in its basic form but that become at times inconceivable when it is called into question the additional package: we go from low figures and flexibility, with monthly, quarterly and annual rates, to a doubling of the total price linked to an annual payment. All without having the certainty that it is really worth it.
Yet we customers, I dare say loyal friends, would be ready to defend big "N" from the attacks of "enemies" who dare to utter vile phrases such as "You Nintendo customers would even buy a piece of cardboard with Nintendo printed on it". Actually we remember well the phenomenon of Nintendo Labo.
Has it been a success? No, but the collectors who

hold such sealed product (which is "aging" at home like a fine wine), are already beginning to circulate it in online stores at murkyly questionable prices. Fetishism? No, simple marketing, a modest laboratory experiment by Nintendo that has sold more than a million pieces. Unbelievable!
So, if we can buy a branded piece of cardboard, if we can watch the Switch selling in incredible numbers, if Mario and Zelda are becoming more and more popular, if by now we are totally loyal... why should big "N" change things? What if a series of titles were discounted on the shelves and such a move was seen as a first sign of weakness, as if we were reaching the end of Nintendo's cycle of success? What if lowering prices was seen as a stability issue for future title releases?
Let's remember that we are talking about the same brands that in recent years have instead been pushed hard to create theme parks (Super Nintendo World): dear friends, we are talking about collaborations with giants such as the great Disney!
We're not talking about discounts or failures. We're talking about a company that's getting better and better every day (or is simply working to make this story believable!) and we're quite happy to pay whole, salty prices so essential to fueling this belief (or reality!).
The collector obviously does not always identify himself as a "supporter", I am a collector and the rules of these big companies are my daily bread. The collector is an opportunistic hunter who calculates daily how best to deconstruct the same financial strategies of the various manufacturers. The goal is to grab a title at the best price of its economic involution, although it is a rare event, the devaluation of Nintendo products, outside the house of big "N", yes, it happens!
Fans obviously are easier to manipulate and often buy sight unseen, not caring much about depreciation.
Nowadays supporting Nintendo has become a synonym of pride and people try more and more not to think about the murky marketing discourse that moves everything. Collectors, on the other hand, are nocturnal hunters, waiting for prices to drop, with shrewd patience, and then jumping on the prey and pinning it down with a sharp credit card and exclaiming "This title is mine too!" and then, after less than a minute, waiting for the economic weakening of the next prey.

::DO WE ACQUIRE PLAYSTATION EXCLUSIVES?

I would be very curious to know how many readers are convinced that Playstation is more powerful than Nintendo. Dear guys, it's not like that! Nintendo fears nothing and nobody! You don't want to believe this? It doesn't matter, but document yourselves, in fact, news fresh from December, Nintendo will soon collaborate even with Sony! Why?
As with "Nintendo Labo", this is a position decided by marketing, useful to collect data and formulate algorithms and theories that will then be fed to the numerous and ravenous mathematicians of the two companies.
A very interesting experiment for these two great houses.
PS exclusives have always been the real ace up Sony's sleeve, managing to sell consoles even for a single game. So get ready: if you love real-time strategy games but you don't have a PS4, you'll be happy to know that 13 Sentinels Aegis Rim is coming to Nintendo Switch!
Particular plot: no spoilers, rest assured, the time frame will be oriented on multiple phases ranging from 1940 to 2100.
Objective: to save the human race.
Released in Japan in November 2019 and on September 22, 2020 internationally, 13 Sentinels Aegis Rim has been a real success especially on Japanese soil: half a million copies have already been sold worldwide.
Will the two firms of Nintendo and Sony finally be together or each other?
Flowers may be born, or they will be thorns. Hopefully, in any case, competition will be born, to satisfy our hunger for collecting video game artwork!

That's it dear readers, how many have stayed in their own enclosure?
Have we really identified the real enemies and the real friends?
I hope in any case that this article has given you new insights into the intricate world of Nintendo, which for years has fascinated me like a dark detective movie. See you next time and Happy Holidays to all of you!

Retro Interview, discovering new talents: **Erika Wittmann**

by Ermanno Betori

In this issue of Retromagazine we continue the series of interviews dedicated to get to know the new talents that as retro users are dedicated to keep alive our PRECIOUSSSSS.. (Smigol Mode on) :-).

Today we present Erika Wittmann that as we will see is a very active person in the world of demo scene and this is her interview!

Hi Ermanno,

Thank you for your invitation for this magazine!
My name is Erika Wittmann and I was born in Hungary in 1983.

My first computer was a ZX Spectrum that my dad got for our family when I was around 3 years old. Me and my dad loved to play space shooters and 2D arcade games from very early on, so this is how my journey started. I never had a C64 when I was a child, but due to visiting some friends I have vague memories of some nice melodies in a few games they showed me :)

Going forward in time, I was about 12 years old when we got our next computer, which was a PC, specifically a 486DLC. This was also the time when my life changed significantly because I got hooked on music like never before. I got familiar with current bands and their tracks before everyone else and could talk about them endlessly! Another big thing was the first megamix I got my hands on - I was blown away by how much of a party-feeling it can give you when the tracks are blended with each other, so I became a mix-fanatic as well.

I dreamed about being a DJ and a bit later, both in elementary and high school I became responsible for the school radio. I always managed to record some tracks to tape ahead of their time from some lesser-known radio stations, so I could introduce them to my schoolmates.

Music became my obsession and it took just a little while to recognize that I don't only want to listen, but discovered an inner drive to compose as well - this took me back to the computer. My first song was made in general midi, in a program called Midisoft Recording Session, when I was 13, combined with a simple Yamaha PSR-2 synth. Shortly after that, the game changer was when I found my way to trackers with samples - Scream Tracker 3.21 and Fasttracker 2! Got a Scream Tracker manual, where I learnt how to use Fasttracker ;) And with the trackers, I got some module libraries as well that blew my mind completely (for example Purple Motion's Satellite One, Starshine or Skaven's 2nd Reality).

Also, in these modules, the composers were all mentioning some sort of parties and I got very curious.

I must have been 14, while attending high school, when some posters appeared on our school wall, promoting a computer party called Jumper '98 with competitions in my hometown, Debrecen. I didn't know what to expect when I entered the event with floppy in hand with my music compo entry, but what I saw was beyond all my expectations. Some people were doing the same things I was, like composing on trackers, some were working on some graphics or animations, and sure, some were just fooling around gaming, but I immediately felt like I arrived home. In fact, so much so that without any hesitation I started to chat with random people asking them about their stories with tracking for example. I didn't realize until then how much of a male-dominated scene it is, but by that time there were a few girls around as well - coders, graphicians and also some musicians -, and it didn't take me long to find them either (still very happy when I see girls joining the scene - never let yourself talk out of any tech-related hobbies, girls, if you have a genuine interest!



Fig. 1 - Erika Wittmann

Doesn't matter how they want to push you down, you can do it!).

So, this was my entrance to the demoscene, and after the first party I quickly joined IRC and started to get to know the community as well. I begged my parents to let me go to the parties that followed - It took some work but luckily I succeeded! :) Whenever I could, I tried to enter the compos as well.

The Demoscene became an essential part of my life. A few years later, at a New Years Eve party of the group Inquisition welcoming 2003, I got my hands on a CD with a few tracks that I had never heard before - another milestone - my first collection of C64 Remixes! Honestly, before that I wasn't a big fan of chipmusic, but getting these made me understand the beauty of it completely, since it didn't take long after this to find my way to the original SIDs. The unique sound of the SID was mind-blowing for me.

From this point, C64 Remixes and SIDs were a constant part of my life. They came with me everywhere, and all my harddrives had a directory for them. Found my way to remix.kwed.org (RKO), remix64.com, c64audio.com and SLAY Radio (the radio playing Commodore 64 remixes) shortly after that.
I wrote my first C64 Remix in 2008, but only uploaded it to RKO 4 years later.

Meanwhile I got my first job as a webdesigner when I turned 21, because before that I did a few web-related hobby-projects, starting with Flash animations. I have worked professionally as a webdesigner since, making it about 17 years now, and occasionally I also do print and other type of graphic design related things as well (for example the Remixer of the Year Award trophies over the past few years ;)) I was drawing my whole childhood, I guess this also gave some foundation for the field, but unfortunately I don't do that very often anymore.

The idea of creating mix-sets out of C64 Remixes was hovering around me for a long while, and following up a successful remix project made for Árok 2016 (the Hungarian 8-bit Party) I finally joined SLAY Radio's IRC channel to find some more people to talk to about remixes. After I appeared, Slaygon sent me a friend request and I reached out to him to ask if I can upload megamixes to RKO (funnily at this point I had no idea who he was other than an organizer type of guy) and he said that would be rejected, however such things are always welcome on SLAY Radio. I thought this could be something, and we started to talk about broadcasting as well. My first show was an interview podcast about X-2016 (currently the biggest C64 party in the world) and after a realtime test, I got my broadcaster

account and got going at the start of 2017.

I broadcasted on my own for a few months, and then I started to think that it would be even more interesting if I could have a conversation with someone else, also I was more than excited to interview some of my favourite composers / remixers that I had been listening to for ages, as well as some SLAY community members. And of course, the rest is history - even working remotely together with Slaygon, we realized we have way more in common than just the C64 Remixes. After moving in together I asked him to come back to broadcasting to co-host the shows, and I am more than happy he said yes! Since then we do the liveshows together on Thursdays and he dove deeper into 3D animations and videos, so the radio has become a true audio-visual experience thanks to him!

My dream of DJ-ing with C64 Remixes also came true, because on a few Árok parties and Back in Time Live events in Norway I did some sets, and the latest one just happened over the last weekend on Transmission 64, which was a blast, and hope I managed to forward some happiness with it to everyone else too!

Goals for the future at this moment are creating more DJ sets on a regular basis since I enjoy it a lot, making music a bit more often (I make some remixes and music compo entries for parties mostly because for me, a deadline is a good push), I might go back to do some SIDs as well (I have done one so far, so only scratching the surface yet on that department) and of course keep going with the SLAY Radio shows and our weekly schedule, hanging out with all the lovely people that surround us! If you want to join the chatter on our Discord, you can do it here: https://slay.radio/discord

Keep up the creative spirit people, and Keep Da Scene Alive! And thank you guys for all your work with Retromagazine!

You can find most of the things I've done via the links below.

My Demographic Story:
https://demozoo.org/sceners/26874/

SoundCloud & MixCloud:
https://soundcloud.com/erika-wittmann
https://mixcloud.com/erika-wittmann

Profile Remix64:
https://www.remix64.com/member/ziona/

SLAY Radio:
https://SLAY.Radio
https://youtube.com/SLAYradio
https://twitch.tv/SLAYRadio
https://www.facebook.com/SLAYRadio
https://twitter.com/SLAYRadio



Fig. 2 - SLAY Radio logo

Armalyte for Commodore 64

by Mic the Biker Novarina

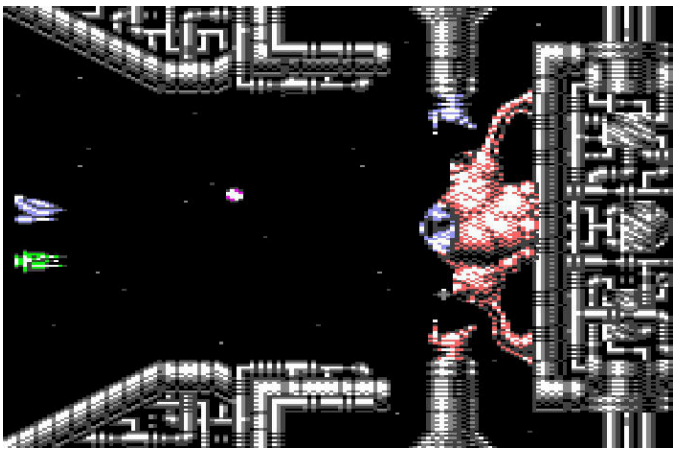
Hello Retro-maniacs, Mic the Biker takes you back in time with Armalyte. You know that I love Shoot em ups of a certain type, and it's a kind of mission for me to give the glory it deserves to a genre that has made the world of videogames great. In their space incarnation, they marked the golden age of videogames, in the arcade as well as at home. After Delta's review, which appeared in the RMW ENG #10, it's time to talk about the second chapter of the saga. In fact, Armalyte was marketed by Thalamus as its sequel: both are horizontal scrolling shooters, but another programming team created Delta.

Talking about videogames is perhaps the only way to get us fans more or less in agreement. We find this colossal shooter inside TheC64 machines, big and mini. We find it everywhere to be able to emulate it and some lucky ones may still keep the original. However, the important thing is that when a game is good you play it, the rest is unuseful. The "where you load it" fades into the black, the quarrels about what is better between original, re-edition or emulation disappear: a good game makes everyone agree.

Armalyte is a horizontal scrolling shooter developed by Cyberdyne Systems in 1988. I remember as if it was yesterday, that day in early December when I went to a shop in the center of Turin to buy it. I saw a review that spoke excellently about it. I did not sleep at night Just looking the photos. Here I am, with my ten thousand lire in hand, the offering to pay for a strictly pirated tape copy. The journey back home on number 33 bus seemed never ending. Even the full volume Testament metal music on headphones didn't help to distract me a bit. I was wondering if those photos seen in the newspaper were true. Finally arrived home I ran into my room and, after turning on the big biscuit, I started loading the game. Time seemed to stand still. It was no marked by a clock but by the datasette's counter. The patience was fully paid: Armalyte, for myself, was, and remains today, THE shoot em up on Commodore 64. Like any good game of this genre, the dynamics are Einstein proof. The aim is to advance to the end of a long horizontal scrolling level where the screen will stop and the battle with a big boss begins. Beating the Boss allows the player to advance to the next level, for eight different worlds. We will meet smaller bosses in the middle of each level and these must be killed to progress further.

The graphic is impressive

Graphically the game is impressive. He manages to take Delta dynamic and raise it to a different galaxy. The atmosphere is more than sidereal: the metal effect is so cold that it can almost be touched. There is an impeccable fluidity in scrolling, with excellent parallax on three levels. We find, compared to its predecessor, much more graphic variety in the levels. The sprites are filled with animation frames that give an almost hypnotic movement effect and an obsessive attention to detail is generally found throughout the game. And the bosses at the end of the level leave you speechless, a feast for the eyes.



Top audio with Mr. Walker

A "monster composing" music artist, Martin Walker, composed the audio tracks. The music in the titles is spectacular to the point of leaving some regret in not finding it in game. The sound effects are however flawless. The difficulty in the game is almost perfect. During each level, the player encounters a lot of enemy ships, many of which fly in formation. You can upgrade your weapons through an innovative system. A great difference with most previous shooters is that any upgrades gained during a level are not lost if the player's ship is destroyed. This is the keystone that gives a new concept of playability. The frustration of losing everything in a place where without power up you are f***ed, disappear. As a result, the life of the joysticks increased as I no longer threw them against the wall. Only at the beginning of the next level does the ship lose all upgrades, except for batteries and generators. However, even here, the programmers' work has been divine: the difficulty is recalibrated to keep



you alive, at least until, the battleship is upgraded again. The main player's ship starts the game with a forward firing laser that produces two shots and a battery that powers the most powerful weapon. There are three types of this most powerful weapon. They all fire forward with the following differences.

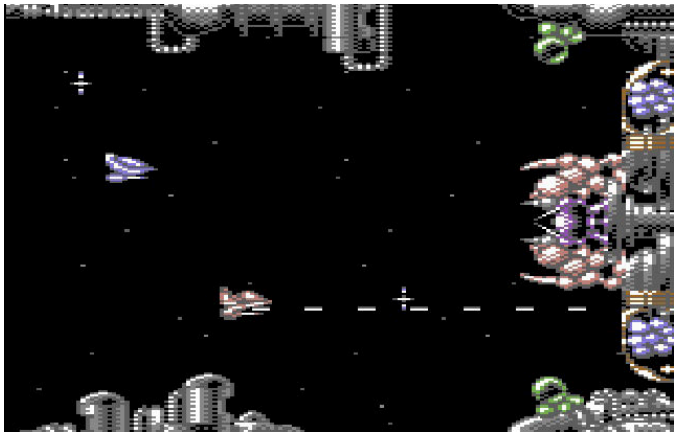
Type A: A long, sustained blast that can pass through the scenario.
Type B: Releases several small laser beams around the spacecraft.
Type C - similar to Type A, only the blast is much shorter and does not go through the scenario.

Firing these weapons lowers battery power, with Type A being devastated but consuming more and Type C less.

One, but also two players

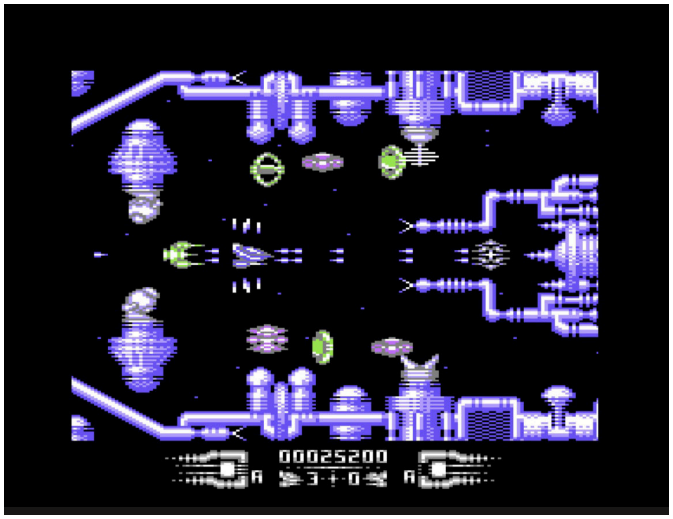
There is also an automatic drone ship in the single player game. It has the same capabilities as the main ship and follows it around the screen. We know, R-Type has set the standard but here the dynamics are different. In the two-player game, a second red ship, with a different design, replaces the drone. Yes, you read that correctly. Second ship, two players! This ship is completely controllable by the second player, at the same time: not many shooter games of this genre were like this. This will take a high playability, and longevity is multiplied by two!

Several power-up slots can be found throughout the game which, when hit repeatedly, turn into a variety of power-ups in sequence. Here is a list, I do not remember if they are all or someone I have forgotten. We have Extra Forward Fire, which increases the number of forward shots from two to four. Very useful is the Tail Fire, which adds shots fired to the rear. Then there is the vertical cannon: it, as the name suggests, adds the vertical shot. I loved the trident - it adds two side guns to improve forward firing speed. Then one that adds more shots to the guns, creating





a wall of fire. We take a look to the slots for the super weapon. There is the generator, which increases the battery-charging rate and the battery itself: it adds one cell to the ship's battery, up to a maximum of four. This means shooting hard, very hard. You can collect power-up by running over them. If taken without being converted into an upgrade, the ship becomes invulnerable for 5 seconds, and I assure you that it is often very useful, such as in the third level where there is a large amount of upgrades in a dead end. If he did not get an empty power up, he would be certain death.



The controls are excellent, very precise. The handling of the fire button is great. A short press of the joystick button fires the standard lasers and, holding the button for a longer time, the super weapon. To switch between the three types, press the Commodore key on the keyboard. When playing in pairs, player two will have to press "?" to achieve the same effect. In solo mode, the drone ship can be controlled by pressing the space bar on the keyboard. The default mode sees the drone ship follow the player's ship around the screen, but pressing the space bar locks the drone ship in its current position. Also during the game pressing Enter will pause while Q will end the game.

Biker reflections

Critics and almost all magazines of the time acclaimed Armalyte, receiving a multitude of awards. Gold medal on Zzap! with an overall rating of 97%. A very high rating for a shoot em up. It was SuperStar in Commodore User magazine, while C&VG rated it Hit of the Month. Computer Games Week magazine gave him the FAB award and a 93% rating. And as a solemn commendation best 8-bit graphics of the year at the golden joystick awards. Dear readers, Armalyte is forever. It was the golden age of shooters, and now you can find this title everywhere. Emulated, original or inside the new big or mini machines. The only advice I can give you is to play with it. Because such a masterpiece makes everyone agree.

Galaxian, from 1979 to 2021

by Mic the Biker Novarina

The first thing that happened to me, when I read the news on the Arlagames website, was humming with a smile the tormentor that marked the beginning of the new millennium: "Oops ... I did it again". Exactly, the one in which Britney Spears appears in a skinny red latex pants and that, even if the song sucked us, the video was always good to watch. Well, the Arlasoft guys, after Galaga's magic work for the C64 (and which we talked about in issue 33), also decided to convert Galaxian, a milestone from 1979 and an absolute precursor of that kind of Shoot em Up. Yeah, "They did it again".

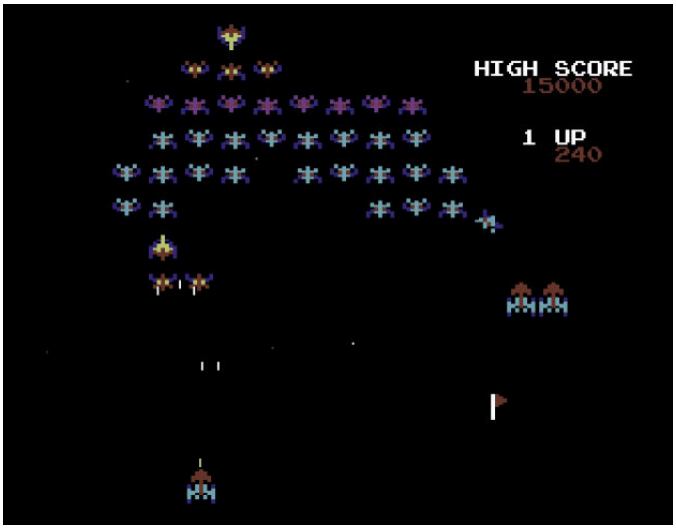


I don't think Galaxian needs any introduction, everyone knows that we will have to eliminate the alien swarms trying to make a Top Score. Logically we will have bombs and missiles to shoot and to avoid the kamikaze enemies. But Galaxian, probably due to 1979 hardware limitations, requires planning and strategy. In fact, we can only shoot one bullet at a time: to fire again we will have to wait that it leaves the screen. You immediately understand that a thrown shot, which does not hit anyone, forces us to dodge bullets and enemies without the possibility of doing anything else. These swarms move from side to side, and shooting enemies in formation is the quickest way to clear the screen. Unfortunately, this way made less points than hitting a single enemy while gliding, but it is one of common tactics to move forward in our mission. As we clear the swarms, the enemies will dive faster, simultaneously with the attack of the others. Therefore each phase raises the difficulty level if compared to the previous one.

In game, the swarms color, besides of giving a fantastic visual aspect for that times, had its meaning. Blue aliens move slower and have less sweeping movements which make them easily to destroy. They are dangerous when trapped on one side of the screen, as those in the middle row move faster. Purple aliens move faster and have sweeping motion, allowing you to shoot from the opposite side of the screen. The red aliens will attack together with a flagship, imitating its movements and protecting it. Sometimes can happen that they attack alone, and their speed is intermediate between blues and purples. The yellow flagships are the ones to shoot down to get a big amount of points and to calm the swarm. We have seen that as you goes on with the game, enemies attack in greater numbers and can drop more bombs. However, destroying a flagship calms the swarm for a while, allowing you to kill them easier. To get the maximum points, first we will eliminate the red escort aliens and then the flagship: doing so the mothership will be worth 800 points, which is a lot 'cause the game has a very low scoring mode. Watch out for aliens flying through the bottom of the screen, or leaving the side of it: they will join the swarm from above. Therefore, an unescorted flagship will quickly leave the screen to take the top row.

Galaxian and his conversion

First, we have to say that there's already an official conversion of Galaxian, dated 1984, and it wasn't bad. Then you know that there are countless clones released over the years, but we are here to talk about this new conversion. Arlasoft, after finishing Galaga, had the idea, given to them by a Facebook post, to convert Galaxian. They thought it would be easy to convert Galaga and turn him into his predecessor. It only took a couple of hours





to transform him into Galaxian, but playing the arcade version the boys noticed that the enemies' movements is very different. It quickly became clear that the only way to reproduce it authentically was to tap into the original arcade code. We read on their site the exciting conversion phase of the game. Scott Tunstall did a great job of reverse engineering with the code. Despite not having full knowledge of the Z80 code, he was able to convert it to 6502 for the C64. The algorithm that controls enemy attacks is a short but ingenious routine that Scott himself admits, "he doesn't fully understand", but it results in varied and fluid attacks. In order for the PAL version to run at the same speed, some parts of the game run at 60 fps.

With Galaga code as a base, it only took two weeks to go from concept to first release, including several improvements borrowed from Galaga, such as a high score table, shooting stats, and 2-player co-op. players. Yes, you read right: even in Galaxian two players can play together. The mode is the same of Galaga: they play together by sharing the three lives. Teamwork is essential to progress through the levels, but beware that the two players compete for points. The player who is currently losing will lose the last life. Also very nice is the Speedrun mode, which consists in having to finish the first game session in the shortest time possible. With the help of testers Jay Bell, Eric Akeson and Stepz, the game has been refined and is now closer to the arcade experience, probably even closer than Galaga. Compared to the latter it is certainly much more difficult and requires patience: you have to learn the right moments to attack and those to avoid it, looking for problems and be patient. Obviously, in certain situations, you can see the limits of the Commodore machine. Sometimes there will be an inevitable flicker when large amounts of enemies are together in a portion of the screen. The movement of the swarm takes up many more CPU cycles than in Galaga, so on the NTSC version will have some flickering sprites, and fewer animated stars: they will be moving on a frame only if there are cycles available to do so.

Biker reflections

What about guys, "Oops .. they did it again", and the result is even more amazing than the one obtained with Galaga. It is good to play even on smartphones, I leave you to imagine with a Joystick in hand. The single shot forces us to be more precise and to think about where to shoot and when. In two players mode, it is practically infinite, its longevity reaches the good old 100% of Zzap memories. Of course, this game can be downloaded for free here: https://arlagames.itch.io/galaxian-dx-c64. If you want, you can make a free donation, which these guys absolutely deserve. In our life we have thrown away lot of money into boxes of doubtful origin with authentic filth inside, at least here we are sure that we have a masterpiece in our hands.

ListAmiga - The 5 most overrated Amiga titles!

by Giuseppe Rinella

Welcome back to all of you, friends who, like me, waited for Santa Claus just to make a double together with one of the thousand masterpieces on the Amiga, thus sabotaging the delivery of gifts to the rest of the world!

During the Christmas season you know, we are all better, we exchange gifts and on holidays we eat meals that are not measured in courses, but in hours.
The list I want to offer you this month, however, goes slightly against the grain of all those nice sugary feelings. It is a list decidedly unflattering towards the titles I am going to discuss, it is in fact de:

THE 5 MOST OVERRATED AMIGA TITLES

It is necessary to make a premise, making a quick trip back in time, going back to the roaring early 90s. The days when the undersigned, much younger, spent the little money he had on video game magazines.
It goes without saying that at that time there was no trace of the Internet, so in order to be constantly updated on our beloved video games, these magazines were our only resource. Magazines that then had the great power to create enormous expectations, often confirmed by reviews to say the least enthusiastic.
In some cases, however, all this enthusiasm was completely unfounded or almost, to the point that a few times I wondered if the reviewers and I had tried the same game.
It would be unfair, however, to put all the responsibility only on those who wrote video games, as we will see in some cases, in fact, huge expectations were created by other "actors" in the game.
These are not bad games, at least not all of them, but titles announced (or reviewed) as masterpieces that, however, once tried, revealed not to be at all.

Another rightful premise: these are obviously personal opinions, for at least one of the titles that I will discuss I risk lynching, I think even two, I am aware!
If you don't agree with me (but even if you do), visit us on our social channels and tell us your opinion!
We're off!

ZOOL

I start with one of the most representative games for Amiga, not so much for its quality, but because it is one of those titles that any Amiga owner has played.
Ever since they started talking about Zool, a big mistake was made, it was presented as the answer to Sonic and Mario, Zool would then become the Amiga mascot, as well as the best exponent of its genre on the Commodore machine.
Just writing a sentence that included Sonic, Mario and the title of a game, at the time generated a hype of biblical proportions.
Up to that point, however, the "merit" of this choice was Gremlin, a pure marketing choice of course, and it worked very well. It caught the attention of any Amiga owner, generated interest and high expectations. It also collected an important sponsor (Chupa Chups), something never happened before for an Amiga title, in short, it was a communication success.
Coming to the game itself, since the previews Zool was described with an infectious enthusiasm, finally even Amiga would have its leading character, the protagonist of a true masterpiece.
The review finally arrived, watch out for the "the" article, not random at all.
Yes, because the first ever review was published on The Games Machine of July/August 1992, but Zool was published in October of that year. That the editorial offices of the specialized press received copies of the games in advance is obvious, it happens even today, but this was far too early.
The truth is that the review was written trying a demo, demo that also happened in my hands and yes, it promised very well, but remained an incomplete version.

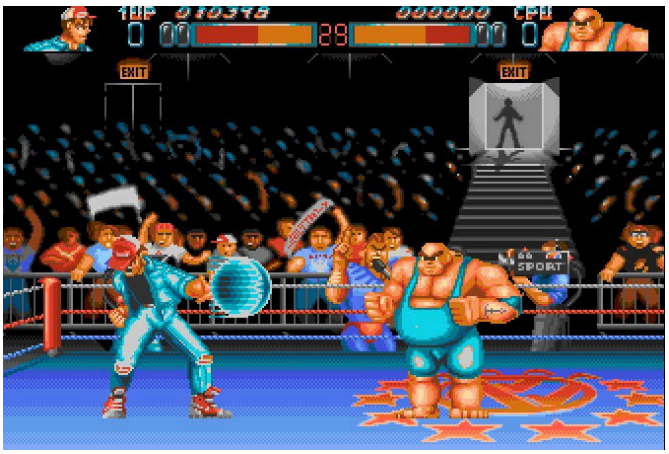


Zool

The final score was definitely high and comparisons to Sonic were wasted.
A few months later the game finally came out, beyond a little too early reviews, how was this Zool? Zool was...how should I put it...meh.
It was not a bad game, mind you, technically it had nothing out of place, graphics and sound were very good, everything flowed without any particular uncertainties. As for the gameplay, there was everything that should be in a platformer: extended levels, power ups, bonuses, mini-games and end-of-level bosses.
The protagonist had a variety of moves of all respect, especially when compared to other titles of the same genre. Zool could shoot, jump doing pirouettes useful to eliminate enemies, climb walls, slide.
Even on the longevity you could not say anything bad, to finish the adventure you have to go through seven themed "worlds" (the candy-themed one, the musical one, tools and more), consisting of three levels each.
I mean, the game was good and didn't seem to lack anything, so where was the problem?
The problem with Zool was those sky-high expectations, magnified for months and largely unmet.
The truth is that Zool, as we would say today, did not even tie the shoes to Sonic and Mario, they were on completely different levels, starting from the charisma of the protagonists onwards.
It was enough to play a quick game of Sonic to see how the latter was on a much higher level, from any point of view. I mention Sonic because Zool is mainly inspired by the Sega game, more than Mario.
Even on the Amiga there were platformers (already on the market when Zool was released) that I personally, though different, have always considered better. Fire and Ice for example, a small masterpiece, but also James Pond 2 was (and is) a jewel, released a year earlier.
In short, Zool is a good game, without particular defects and does what it must, which is to entertain and does it very well.
Unfortunately it paid the heavy ballast of announced masterpiece that was assigned to it, in addition to the thankless task of rivaling the leading characters of the great giants Sega and Nintendo.
Characters, however, who were simply playing a different league.

BODY BLOWS

Body Blow represents, in my humble opinion, the most striking case of high expectations not fulfilled, one of those disappointments that I still struggle to process.
There were several factors that contributed to this, the



Body Blows

trade press was certainly one of them but it wasn't the only one.
When Body Blows appeared in the preview pages of various magazines, the images promised very well right from the start. The enthusiastic tones (to say the least) with which the title was described during the development phase, made us hope. Every month, or almost every month, new images appeared accompanied by more and more electrifying articles.
In addition to this, a great contribution to the enormous expectations that accompanied for months Body Blows, according to the writer, gave him a couple of "external" factors, both related to Street Fighter 2.
The first was the release of SF2 coin-op version, which exploded the craze for 1vs1 fighting games. The arcades were soon invaded by clones of Capcom's masterpiece, some successful, others decidedly less so. Shortly thereafter, the same happened on 16-bit consoles, where SF2 was the absolute benchmark, thanks to the monstrous conversion on Super NES.
The fighting fever infected even us proud owners of our beloved Amiga, of course, despite the fact that it was perhaps the least suitable genre to be played on our machines, given the historical limit of the single button (remember that SF2 used six buttons).
The second external factor was, once again, SF2. But this time I'm talking about the Amiga conversion, an epic debacle. The frustration (along with a lot of anger) that ensued only increased the hunger for fighting games, we too wanted our SF2, or at least something that came close.
Into this climate came Body Blows. And it was bad.
Graphically drab to say the least, in photos made a very good impression, but in motion was another story. The characters were animated little and badly, backgrounds were practically static and the choice of colors, which made everything look pale and dull, at least questionable.
There were only four usable characters in single player, in general they were all pretty anonymous, some even

ridiculous.
Playability? Really little stuff, there were no projections (or holds, if you prefer), each character had only one special move that was performed by holding down the button. Between the different characters there was no difference in strength or speed, the effect of all this was a total absence of any strategy during the fights.
On the sound nothing to say instead, excellent voices sampled combined with music by Allister Brimble, a guarantee.
Played in single player it was a deadly bore, played with friends the situation improved for sure, but for the presence of friends, not for the game.
The reviews underlined, in most cases (except on CVG, reading again today the review is almost comical), a certain disappointment rewarding Body Blows with high ratings, talking about it as a title with many flaws but technically remarkable (mah).
In reality, no one wanted to admit to the end how Body Blows was a bitter disappointment and a product that, to be generous, did not go beyond mediocrity.

ELFMANIA

Another fighting game, a troubled genre to say the least on the Amiga.
The story of Elfmania has, in my opinion, a lot in common with Body Blows. Also in this case there were months of previews that screamed miracle, accompanied by images really remarkable.
Moreover, as for the Team17 title, part of the expectations/hopes were due to the fighting fever that Street Fighter 2 gave birth to. Add to that the frustration of the Amiga version of SF2, combined with the disappointment of Body Blows, released the previous year, and you're done.
Nothing for which the Terramarque team was at fault or responsible, it goes without saying.
Elfmania finally came out, and it was an awesome demo!
If we wanted to show our friends what the Amiga was still able to do at the technical level, Elfmania was the title to show, no doubt, but keeping the joysticks at a distance so that the big bluff was not discovered.
Elfmania was graphically a small miracle, on Amiga had never seen anything like it in a fighting game. The scenarios were nothing short of spectacular, rich colors and beautifully animated. Even the famous perspective floor, which has become a standard from SF2 onwards, was on display. In short, they would not have disfigured even in a coin-op.
The characters were animated pretty well, the design was questionable in some cases, but that's probably more a matter of taste.
Okay, the graphics were there, but how was the rest of



Elfmania

it? Yet another disappointment.
The music wasn't bad, on the other hand the ridiculous voices of the different characters were irritating to say the least.
Coming to the playability though, Elfmania showed all its inconsistency due to several factors.
There were only six characters in total, three selectable from the start, the other three unlockable.
The three selectable were quite ugly and with movements at least questionable, if nothing else with the other three the situation improved, but it must be said that getting worse was impossible.
Speaking of real combat, Elfmania bordered on the ridiculous. No grip, no special shot, the possibility of varying the trajectory of the jump in mid-air, as in a platformer to be clear, but in a fighting game becomes a heresy.
You never had the feeling of causing the slightest pain to your opponent, which was clearly a precise choice, the developers clearly wanted to offer something very different from SF2 or Mortal Kombat. The search for originality is always a good thing, but in this case it simply didn't work.
All this, combined with questionable mechanics (the possibility of hitting opponents with the coins they lost once they were hit, for example), made Elfmania yet another disappointment.
The magazines were far too generous in reviewing it, rightly recognizing a technical realization amazing, but masked the cosmic nothing from the point of view of playability and fun. Even in terms of quantity Elfmania was very poor, six characters and six scenarios were really little stuff.
Filed another disappointment was not left that to return, not without a veil of sadness, the excellent conversion of Mortal Kombat

PROJECT X

I can already hear the screaming crowd that, holding pitchforks, wants to punish my heresy.

I repeat, to avoid any misunderstanding, that what I write is the result of my personal opinion, but also of my "history".

I emphasize this second aspect because Project X arrived in my Amiga drive relatively late, this probably contributed to my lack of enthusiasm (to speak of disappointment would perhaps be too much) for the Team17 title.

Project X was published in 1992, a golden year for shot'em up on Amiga, almost at the same time Apidya and Agony were released, so much so that on The Games Machine they were all reviewed in the same issue.

All excellent titles that I had the opportunity to play at different times.

The first of the three was Agony, then Apidya and finally Project X.

During this time I was able to constantly read about how Project X was an epoch-making masterpiece, which only increased my expectations.

When I finally tried it out I had confirmation that yes, what I read was true, so why put it on this list?

Let's start from the purely technical aspect: Project X was graphically excellent, from the environments to the enemies up to the bosses at the end of the level of truly remarkable size, everything flowed smoothly, no minimum uncertainty. Even on the sound aspect nothing to say, music and effects of great level and with Allister Brimble could not be otherwise.

The playability really had nothing wrong, on the contrary. In short, everything was made with extreme care, Team17 confirmed that at the level of programming, few at the time could compete.

The problems, for me, were essentially two: the first, the already mentioned comparison with the two direct competitors.



Project X

As great as Project X's graphics and sound were, I wasn't as impressed with it as I was with Agony. I found the latter much more spectacular and artistically much more inspired, which I still think.

Apidya was also technically top notch, beautiful graphically and a joy to listen to, thanks to Chris Hülsbeck's music.

I found Agony and Apidya more interesting in general, probably because of the less banal settings than Project X, after all piloting a spaceship and shooting alien enemies was nothing particularly original.

Second problem, definitely more influential: Project X had a monstrous difficulty, I would say unmotivated.

The high difficulty is part of the basic characteristics of this kind of games, even Apidya and Agony were not exactly a walk in the park. The title of Team17, however, gave me the impression of pushing too much on this aspect, to the point of making the game frustrating and ultimately not fun.

Here, Project X was not fun for me.

On the subject of difficulty, my opinion has not changed over the years, I think that a high level of challenge is always stimulating, but the line between "difficult" and "inexplicably impossible" is very thin and when you overdo the fun disappears and makes room for anger and frustration, as a result the game loses its sense that must always be to entertain.

I was not the only one to think so obviously, so much so that about a year later was published a version of Project X revised and corrected, with a difficulty certainly not low but definitely more sustainable.

Ultimately, Project X was a title that could boast an impressive technical achievement, free of flaws. But it didn't impress me at all, not as much as I would have expected and not like its direct competitors.

As far as I was concerned, the fun was to be found elsewhere.

AQUAVENTURA

The case of Aquaventura was very different from the titles discussed so far, in fact it was not an overrated game. On the contrary, when it came out the reviews were not flattering at all, some more merciless than others, but in general the reactions were completely negative.

Aquaventura is perhaps the title that on Amiga first generated a sensational hype, not entirely unfounded.

It was a Psygnosis title, then and for many years to come synonymous with extreme quality. Aquaventura, judging by the images that circulated and by what we read, seemed to be yet another spectacular product of the Liverpool studio.

In addition to all this, the long development time did

nothing but feed more and more expectations for this masterpiece announced. Aquaventura in fact was announced in 1988 but saw the light in 1992, after four long years, by the standards of the time it was a couple of geological eras. A period so long that the hopes of seeing it on the shelves began to fade more and more.

In 1992, however, Aquaventura was finally published, and it remained to be seen if it would satisfy us avid gamers. Before talking about the game itself, it is necessary to dwell on the presentation, an animated sequence in ray-tracing absolutely spectacular. Admiring it is possible to appreciate, among other things, the design of the spaceship that we'll have to pilot, a true marvel.

In short, this is one of the best presentations ever seen on the Amiga, not only up to that point but ever.

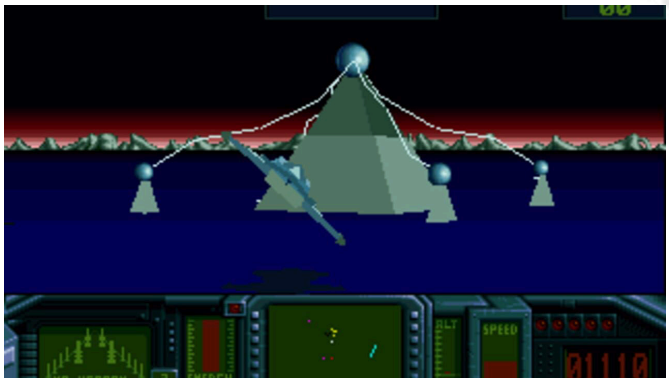
So far, therefore, expectations seemed to have been met. Problems arose once I held the joystick. First of all, what game are we talking about? Aquaventura is a shoot'em up in 3d with a view from the back of our spaceship, at the beginning of each level is made a briefing where the objectives are displayed to find and destroy.

Graphically the game was not bad, it was a mixture of 3D elements (the structures to be destroyed, the sea that will accompany us during all our missions) and bitmap (the enemies to be shot down). Unfortunately, everything seemed slow and decidedly not fluid.

The music was, as per Psygnosis standards, of excellent workmanship. The singular is unfortunately a must and is the biggest flaw of the sound aspect. Throughout the levels the music is in fact always the same, however good it is inevitable that it is repetitive. The sound effects instead are bad, nothing else to say.

Coming to the gameplay, all we'll have to do is find the objectives and destroy them, thus arriving at the confrontation with the final boss, always consisting of a series of spheres. The only difference between the different bosses is their shape, which changes thanks to the different position of the spheres.

The great flaw of Aquaventura, which was deservedly panned by critics, was the monotony that made it extremely boring. The setting was always the same in all missions (except for some color that changed), the music and especially the playability. In addition, we had only one life and once you run out of energy you start over, which is a big problem when the game is frighteningly boring and the idea of starting from scratch again becomes agony. Aquaventura was therefore a mediocre game despite the very long development that made us hope, a victim of the high expectations that he generated and fed.



Aquaventura

Before I take my leave of you, allow me to pay tribute and give endless thanks to Ian Hetherington, who passed away just a few days ago (12/14/2021).

Co-founder of Psygnosis, developer of several games including Aquaventura, he contributed to the creation of some of the titles that have made the history of video games, on the Amiga and beyond, including Shadow of the beast, Lemmings, Wipeout, Destruction derby, Colony wars.

Another sad news, after the passing of David Lawson only a few months ago, also one of the founders of Psygnosis. Two great artists, authentic visionaries that every passionate gamer can not fail to pay homage.

See you again next month dear and dear, have a great holiday season, don't eat too much and play as much as you can.

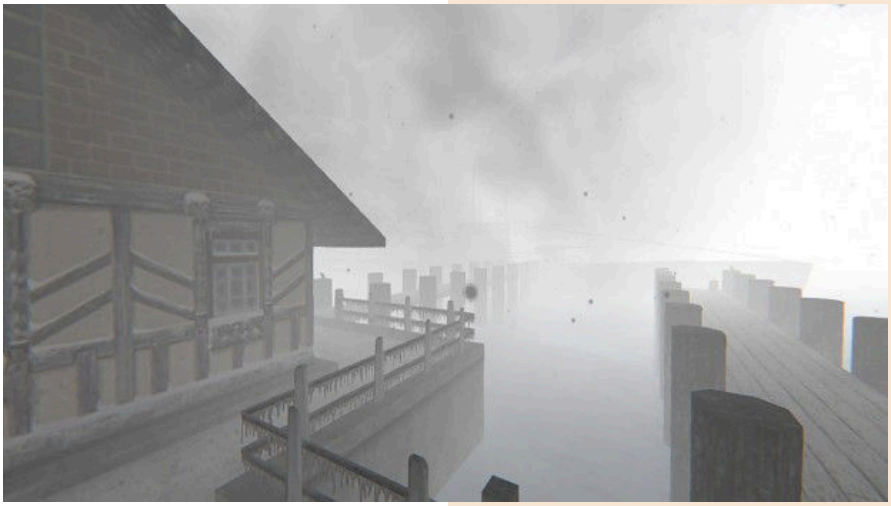
Greetings to you all and AMIGA FOREVER!

NEW GAME

BLADE OF AGONY

Year: 2021
Developer: Team di Blade of Agony
Music: Team di Blade of Agony
Editor: Team di Blade of Agony
Genre: First person shooter
Platform: PC

When talking about Doom you can not ignore the great panorama of modding that gravitates around the title, a landscape that continues to expand over the years and is still growing. We have seen the birth of projects that go beyond a simple hobby, among them we can not ignore Blade of Agony, a mod total conversion for Doom 2 in development for six years and recently became standalone.

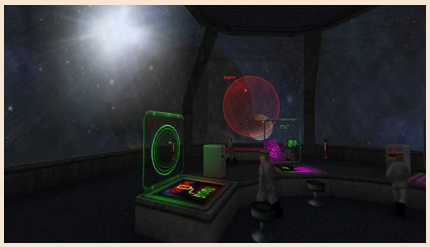


The project sets the game in the Wolfenstein universe, during World War II, proposing classic enemies but creating a totally new world, pushing the Doom graphics engine to its limits.

The game is divided into chapters, which feature a multitude of enemies, boss fights and really huge maps, the whole game can last up to 4 hours and more. There is also a central hub where the player can buy power-ups, talk to NPCs and get new quests.

It draws a lot of inspiration from titles such as Medal of Honor and Call of duty, using audio effects taken from these titles, while maintaining the spirit of Wolfenstein from which it takes the bosses, weapons and collectables. There are also collectible armors, a compass to guide you, in case you get lost in the maps, and a disguise mechanic. As well as special items to collect to complete the missions.





There shouldn't be any problems regarding the hardware to be used but it is still possible to adjust the graphics settings.

by **Diamanti Aharon Maurizio**



OUR FINAL SCORE

» Gameplay 80%
The maps are really big and can be explored, with a play style that can change and make the title repeatable several times, and also present a central hub where you can perform a variety of things.

» Longevity 90%
It can last several hours of play, thanks to the presence of great environments.

NEW GAME

SONIC THE HEDGEHOG

Year: 2021
Editor/Developer: Sonic Team
64
Genre: Platform
Platform: Commodore 64
Hardware: REU 256Kb +

Remember this date:
December 19, 2021.

A historic date for the Commodore 64 and for all the fans of its games. Sonic the Hedgehog's arrival date on Breadbin!



For years there has been speculation about whether we would see the conversion of the two iconic platformers on the 8-bit Commodore platform. In 2019 we got the Super Mario Bros conversion (angering Nintendo in no small part) and now we have the Sega mascot in our hands, in its conversion from the Sega Master System version, and we got it for free.

The latest effort of Mr. Sid, a developer already known for his conversion of Prince of Persia and Donkey Kong Junior, gives us an almost perfect conversion, albeit with the necessary limitations.
Just downloaded and tried we can say that is Sonic in all its beauty. Sprite in HiRes, backdrops taken from the Master System version and a scrolling that will make your jaw drop.

It's all very fast. For all this goodness, however, you need an REU (Ram Expansion Unit).
The REU is a memory expansion that allows fast access to data because it "STOPS" the CPU during its "operations".

Properly exploited (since the VIC and SID are COPROCESSORS) allows to manage the graphics (scrolling in the case of Sonic) in a more fluid way. The REU were put on the market in 1985 by Commodore and since then







OUR FINAL SCORE

» Gameplay 95%
Sega's platformer is a combination of simplicity, speed and unparalleled fun.

» Longevity 95%
It's a game that keeps you glued to the screen... Only with a directional cross and a single button.

you can use (as does for example the GEOS) for this purpose.
An expansion of at least 256kb is required to play this version.

It is possible to do this in emulation by setting the REU inside VICE, or by changing the setting on THE C64 and finally by adjusting the memory through the Ultimate II cartridge on a real C64. Whatever your solution, you will be looking at an excellent product.

The thing I liked the most was how the SID was used. Squeezed in really really well. The soundtrack is perfect and the effects are just right.

Sonic again is very playable and fun. The conversion of the Master System version is a very good choice (for me the Master System version is wonderful and in some points superior to the Megadrive).

The only notes concern the use of REU. At the real hardware level the prices of these expansions are sky high and you have to turn to cartridges such as the already mentioned Ultimate II to play it without problems.

It's Sonic, it's fast... And it's on our beloved C64.

That said, you'll just have to download and play it.

You can download it for free from:
https://csdb.dk/release/?id=212190

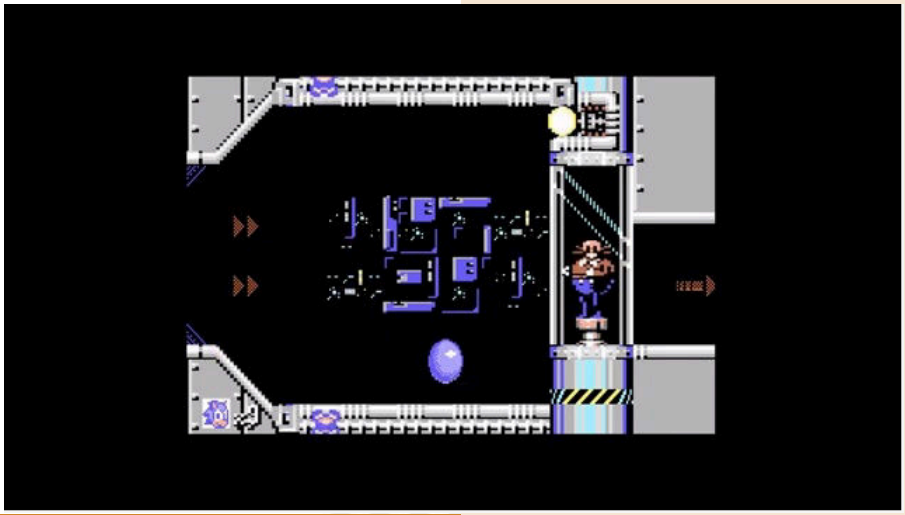


Tested on: Vice 3.5, THE C64 and Commodore 64 + Ultimate II

by **Carlo N. Del Mar Pirazzini**







NEW GAME

SLOW MOLE

Year: 2021
Developer: SlowMoleDev
Genre: Platform
Platform: Nintendo Nes
Website: https://slowmoledev.itch.io/slow-mole

Slow Mole is an 8-bit platformer for NES that has two fundamental characteristics. The first is that it will make us all speedrunners, the second is that it has a level of difficulty that will make us swear and lose patience like few other games, old or new.



It's totally free in its downloadable version and you can play it in emulation or on real hardware (with everdrive). The developer let us know on his itch.io page that soon will be released a "PLUS" version with a few more levels and some new features. But the game on the site is 100% complete.
The entire game is designed around a unique checkpoint mechanism that will only award a checkpoint when the player demonstrates their complete mastery of a given level.



At the wheel of a small gray mole the goal is to survive! On the surface it is the purpose of 100% of the games, but I assure you that in this one is really a titanic feat.
You have two difficulty options to choose from. They both affect the game timer, which might seem insignificant, but as you play you'll notice a big difference.
It's a brutal game. Suitable for players with a strong stomach and infinite patience. The level of challenge is a mechanical nightmare and there will be many times that you will see the little mole slaughtered in one of the terrible and cruel traps present in the various levels.
It's technically polished and also easy to learn mechanics. There aren't too many things to do other than jump and avoid death.

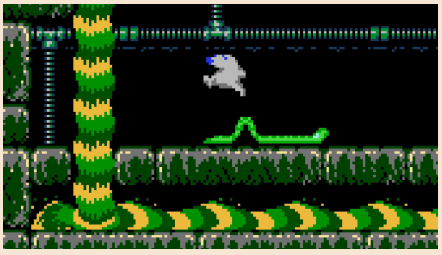


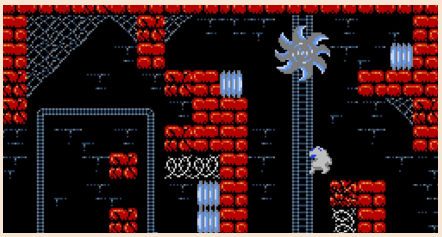
Despite the monstrous degree of challenge, I really enjoyed this title and I must admit that, after the first moments of total hatred towards the developer and his family, I continued to play even dying over and over again.

A title to try for those who like the level of challenge of the glorious old days.

by **Roberto Del Mar Pirazzini**









OUR FINAL SCORE

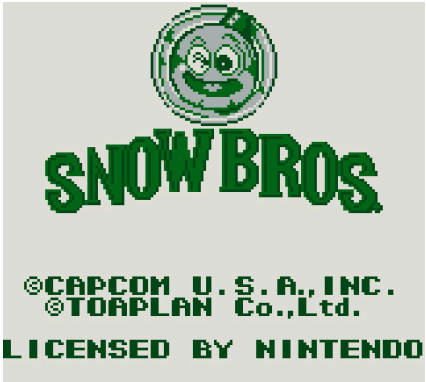
» Gameplay 70%
Easy to learn.

» Longevity 55%
This is a damn difficult game that will often have you shutting down your console and cursing like the worst dockworkers. Not for everyone and not for the not-so-patient.

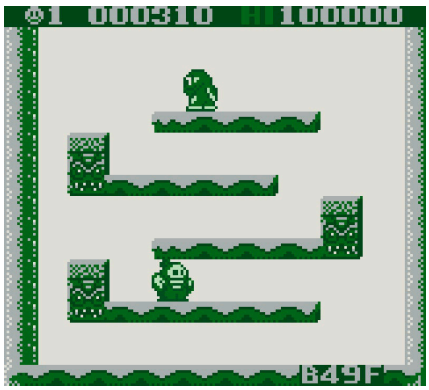
SNOW BROS

Year: 1991
Editor:Naxat Soft/Toeplan/ Romstar/Capcom
Genre: Platform
Platform: Game Boy

Snow Bros. is a 1990 arcade platformer video game, ported to Game Boy in 1991 originally developed by Toaplan and published first in Japan (Toaplan/ Naxat Soft.), then in North America (Capcom/Romstar) and later in Europe.



Starring the eponymous twin snowmen Nick and Tom, players are tasked with traveling through 50 levels, throwing and building snowballs, jumping on and off platforms to overcome level obstacles while dodging and defeating monsters to rescue princesses Puripuri and Puchipuchi from captivity.



Another difference that you find in the Japanese version and the U.S. version is the "food" that gives you, after you defeat the various enemies, in the Jap version is sushi, in the U.S. version donuts, but also money! This Game Boy version is fun and technically well developed.



The game's strength lies in its simple, well-structured gameplay and gradual difficulty.



This is a classic of its genre that you'll also appreciate in this GB version.

by **Barbara "Morgana" Murgida**







OUR FINAL SCORE

» Gameplay 80%
A classic of its genre that is easy to play.

» Longevity 80%
Well-calibrated difficulty even in the portable version.

GAIARES

Year: 1991/2021
Editor/Developer: Telenet, Retro-bit
Genre: Shoot'em up
Platform: Sega Megadrive
Website: https://retro-bit.com/gaiares/

Gaiares is a horizontal scrolling shooter made by Telenet Japan and published by Renovation for Sega Megadrive in Japan and North America between late 1990 and 1991. It is one of the first games with 8Mb cartridges. The title combines the word Gaia (Mother Earth) with the suffix "minus" (Res), since the Earth has been lost to pollution in the distant future.

In the year 3000, Earth has become a toxic dump ravaged by negligent humans who have left it desolate, uninhabitable and dead. The powerful Gulfer aliens, led by the evil Queen ZZ Badnusty plan to harvest Earth's pollution and soil toxicity to create chemical weapons of mass destruction. Earth's survivors, warned of the evil aliens' plans, mobilize to prevent galactic catastrophe.

The protagonist of the game is the space hero Dan Dare (Diaz in the Japanese edition), a skilled war pilot chosen to lead the battle against the alien space empire with his spaceship. The storyline is reminiscent of hundreds of Anime of the late 70's, memories that resurface thanks to the many graphic sequences in the game in cartoon style.

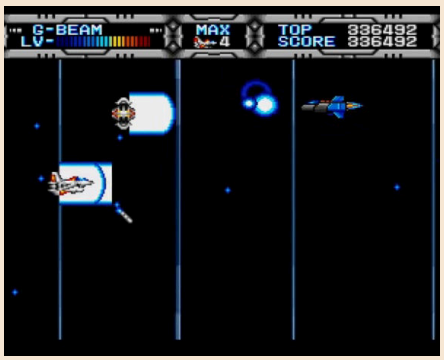
If in most of the games of the genre the ship comes into contact with a capsule to obtain weapons, in Gaires the control device called TOZ (a Drop that always follows us) can be launched as in R-Type and, coming into contact with an enemy, inherit its fire capabilities and upgrade the ship's defense system. The player can then "steal" repeatedly from the same enemy the armament until the strength of the chosen weapon is at its maximum. There are 18 different types of weaponry and each weapon has different strength, speed and destruction characteristics.

As we said the game came out between the end of 90 and early 1991 only in Japan and the USA, missing the European market (arrival only by import). A pity, because it is undoubtedly one of the best 2D horizontal scrolling shooter ever and perhaps the best on Megadrive with Thunderforce IV.

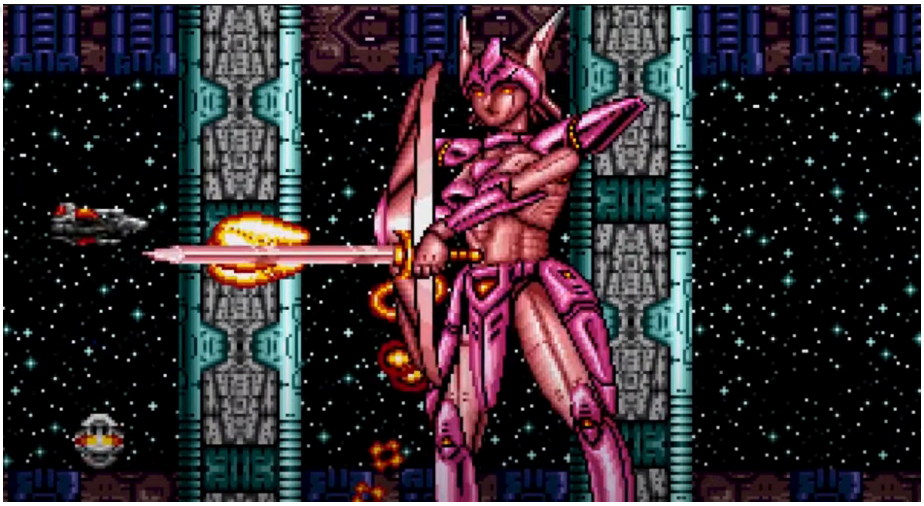
After 30 years, the Retro-bit group has found the courage to reissue it and launch it on the market in a physical version for all lovers and fans. Dispassionate advice? Don't pass it up.









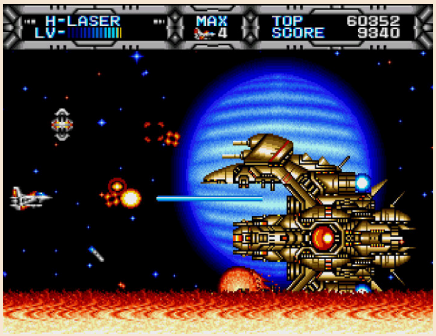


OUR FINAL SCORE

» Gameplay 90%
Innovative and challenging weapons system.

» Longevity 70%
It's difficult and takes practice, but it lets you play and will keep you busy for quite a while.





The game is graphically excellent, fast, frantic and rich in animations and colors. Fluid and without ever slowing down.
It has an atmospheric soundtrack reminiscent of old Japanese cartoon theme songs.
The game system is innovative and the level structure is varied and will never bore you.

It has only one big flaw... it is very difficult! A game that doesn't forgive positioning errors and, above all, asks the player for maximum precision in facing each level and each fight with the Bosses.

At the time MegaTech magazine considered it "among the best of its kind but by far the most difficult shooter on the Megadrive".

It wasn't wrong in its assessment. It's a game for hardcore gamers and not just casual gamers.

The Retro-Bit version features the cartridge, manual, a commemorative t-shirt, a small illustrated book and a beautiful poster inside. All for 50 euros.

Gaiares is an intense game that deserves to be rediscovered.

by **Carlo N. Del Mar Pirazzini**



NEW GAME

METAL SLUG 6

Year: 2020
Developer: SEGA, SNK PLAYMORE, MEGAVOLT85
Genre: Scrolling shoot'em up
Platform: Dreamcast

Metal Slug 6 is a scrolling shooter video game for Sammy's Atomiswave arcade board. It was released in 2006 and is the seventh game in the Metal Slug series. The PlayStation 2 version was released in September 2006 exclusively in Japan. In North America and Europe, the game was released directly as part of the Metal Slug Anthology compilation for PlayStation 2, PlayStation Portable and Wii and Metal Slug Collection PC for Microsoft Windows.



The Atomiswave arcade platform is seen by many as a spiritual successor to the Neo Geo MVS, being a collaboration between Sammy, SNK and Sega. Interestingly, it was very similar to Dreamcast hardware, even more so than the Sega Naomi. Unlike Naomi, however, Atomiswave was largely overlooked due to its rarity and relative inability to successfully break into arcades.

The game is set during the Martian War seen in Metal Slug 2, Metal Slug X, and Metal Slug 3. A new hostile alien race is introduced, the Venusians, who via a meteorite have taken up residence in the bowels of the Earth. Cannibalistic little monsters, with a queen as their leader (known as the Venusian Queen, since her name is unknown), have already attacked Mars and their intention is now to infest the Earth and expand throughout the solar system. Initially the protagonists will face the Martians and the Rebels, but from mission 3 onwards there will be an alliance between the normal army, the Rebels and the Martians. At the end of mission 2, General Morden will tell our heroes what happened to him after saving Rootmars (the leader of the Martians, the final boss of Metal Slug 3); from that moment on, the protagonists ally







with Morden and the Martians to face the new enemy faction.

All the "historical" characters of the Metal Slug series appear - Marco, Moth, Fio and Eri - along with Ralf Jones and Clark Steel from The King Of Fighters and Ikari Warriors series. The levels are 5, each of which provides at the end a boss to face: the bosses of the first two levels are military equipment of the Rebellion, the one of the third is a Venusian robot, the one of the fourth is a biomechanical Venusian creature, the last boss is the giant queen of the Venusians, who once defeated will die smashed.

The lack of this little gem on Dreamcast has been filled thanks to Megavolt 85 that has included it along with a series of titles called Atomiswave Games and is definitely faithful to the original coin op.

The Sega machine does its job perfectly and puts before our eyes a game identical to its arcade counterpart.

Graphically perfect, wonderful sound and a game structure broken and fun. Metal Slug in any of its incarnations deserves to be within any library play and this Dreamcast version of the sixth chapter is a gem.

Try it out. It is available in physical version but also in iso/cdi version to be inserted on a blank CD, on SD or, tested, on Emulator.

by **Roberto Del Mar Pirazzini**

OUR FINAL SCORE

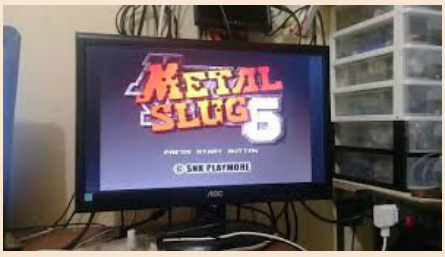
» Gameplay 95%

Metal Slug is fun, cackling, with a level structure broken down from chapter to chapter.

» Longevity 90%

The challenge is always the same of the series. You must have a lot of willpower because the difficult curve rises to the top.









NEW GAME

TECMO SUPER BOWL EDITION 2021/22

Year: 2021
Developer/Editor: Tecmo/ Tecmo
Genre: Sport
Platform: Nintendo NES
First issue: 1991
New issue: 2021

American football entered my veins back in 1987, the year I saw a game on television for the first time (with commentary by Bagatta and Peterson), the same year I went to see the team of my city (the legendary Cobra Imola). I immediately became a fan of the sport and many years later I also began to play it (and I still do, it's been 20 years and more of playing, Ed).

Tecmo Super Bowl was originally released for the NES in 1991. It was the first sports video game to have licensing privileges with the NFL and NFLPA. This allowed programmers the freedom to use player names and attributes, as well as team names, logos and colors. All 28 teams from the 1990-91 NFL season were present and represented here, as well as one of the most revolutionary players to ever walk on the face of a virtual console, I'm talking about the great Bo Jackson.

The game has, in the decades since, created an immense fan base. So big that it has released editions that are always updated, arriving at the latest edition released on the NFL season 2021/22.

There are several modes in the game. You can make a pre-season game/ exhibition, there is the possibility to start a regular season, play the pro bowl and set the teams.
Also we will be able to decide whether to participate in the game as a player/ coach or just as a spectator.

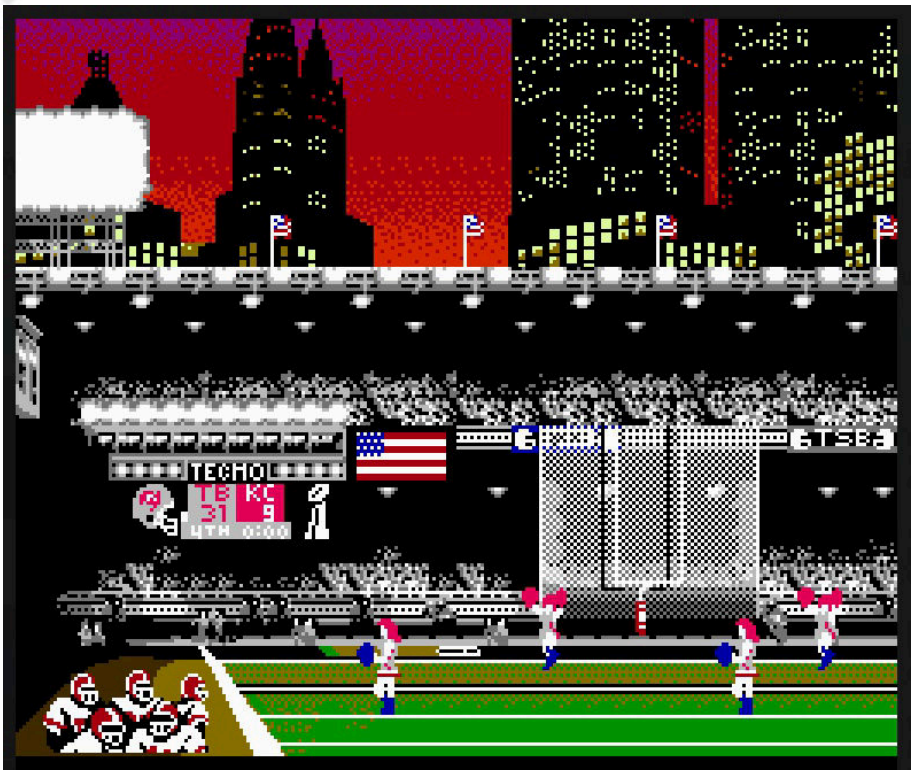


The gameplay is in line with the arcade sports simulators of the past. There are no penalties in the game, which helps keep the pace lightning fast. However, players are prone to injuries and fatigue. Of course, play calling is present. On offense, the playbook consists of 8 plays; 4 pass plays and 4 run plays.



Same thing happens when we play team defense; again we are faced with 8 plays that are used to repel the offensive phase of the opposing offense. If we guess the opponent's calls, our defenders will be able to block the opponent's play coming even to tackle the opponent's quarterback (sack). The playbook can be modified before any game.





You can change the default plays, but you must always keep the 4 passes and 4 runs.

If you can't close the fourth down there is the punt session with the respective punt team and punt block team. Of course, it is also possible to kick the ball for a field goal.

The updated game has some graphic improvements in the colors of the teams and a greater AI of the opponents (some teams are really tough).

The technical aspect is minimalistic but effective and does its job very well.

If you are a fan of American football and do not like the absolute technicalities of modern titles, you will love Tecmo Super Bowl in any of its incarnations.

I'll leave you now, I have to finish the season with my Vikings.

by **Carlo N. Del Mar Pirazzini**

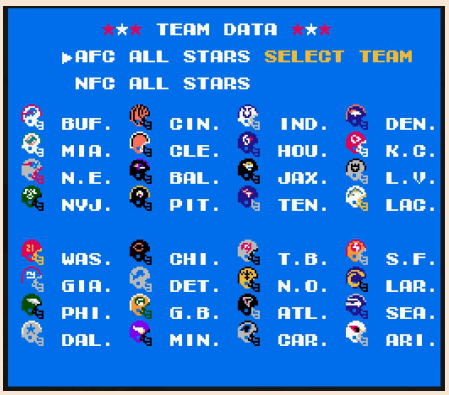
OUR FINAL SCORE

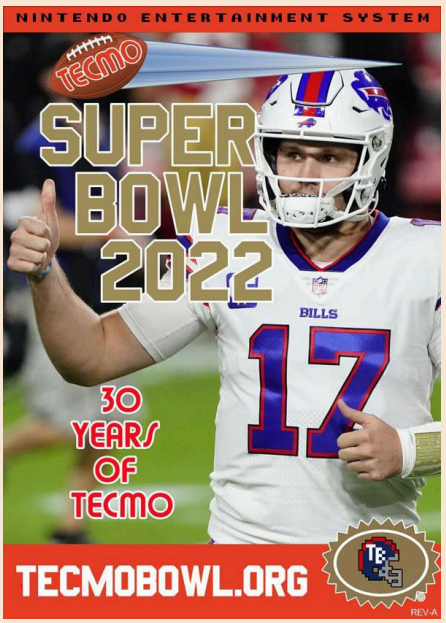
» Gameplay 80%

The game is simple to handle and understand. Everything is played with the directional cross and the two buttons. Simple and effective.

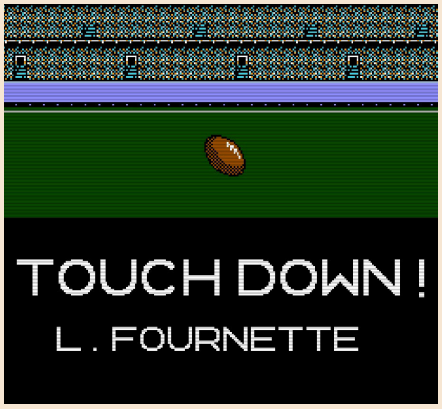
» Longevity 80%

Lots of game options and the NFL season really lasts a lot.









JAZZ JACKRABBIT

Year: 2002
Developer: Game Titan/Jaleco
Genre: Platform
Platform: Game Boy Advance

Finally mine! How many years have I waited! The first time I saw the game play I was so impressed and wanted to have it right away! I finally got it after so long.



And then the combination "game with cute character/tough plus galaxy background" fits right in!
It's a gba exclusive title (after its birth in the freeware world on pc, Ed).
"The game begins when Jazz, during a routine mission, is captured by the Chameleon army on their home planet. After escaping, Jazz decides to retire from his job, but is prevented from doing so by R.A.B.T. HQ, which gives him a new mission to investigate a Saurian attack with the promise of a good cash reward. Jazz discovers the involvement of the Turtle Army itself behind the attack and, following them, discovers that his old nemesis Dark Shell, whom he thought was dead, is seeking revenge on him for his previous defeat."
Jazz Jackrabbit is a 2002 platformer game developed by Game Titan and published by Jaleco under license from Epic Games.
The game style is very reminiscent of modern run 'n gun with the ability to move the gun by pointing it in the eight directions of the pad.

Equipped with a good audio and graphics, the game is really fun and deserves your attention.
Play it please, it's great!

by **Barbara "Morgana" Murgida**









OUR FINAL SCORE

» Gameplay 85%
Simple to "handle" and well structured in play style.

» Longevity 90%
This title brings back multiplayer mode, with 11 unique levels and support for up to four players via GBA cable connection. Only one copy of the game is required to enable multiplayer mode. Incredible Longevity.

NEW GAME

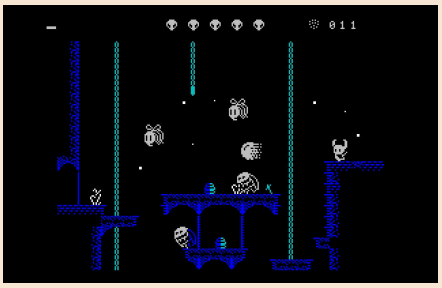
HALLOWED KNIGHT

Year: 2021
Editor/Developer: Jonathan Cauldwell
Genre: Action/Adventure
Platform: ZX Spectrum 48k
Website: https://ejvg.itch.io/hallowed-knight









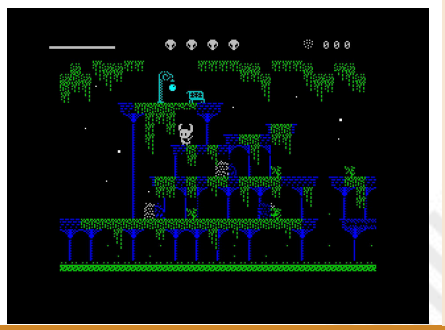


Hallowed Knight is a metroidvania-style action-adventure game for the ZX Spectrum 48k.
Created to participate in ZX dev 2021 using the MPAGD 7.10 engine, as a tribute to the original Hollow Knight, a video game masterpiece available on different platforms such as PS4, Switch or Steam.
The game takes place in Nidosacre, a desolate, underground kingdom where a strange "corruption" has infected everything. Our little hero wields a sword called the "PUNCH" and can also use different spells when he accumulates enough Anima (the bar in the top left corner during the game).
The game is completed well within about an hour. It is possible to play using either the keyboard (recommended method) or the joystick.

The default keys are:
Left-O
Right - P
Up / Jump - Q
Down - LA
Fire 1 (Sword) - M
Fire 2 (Spells) - SPACE

In any case we can redefine them and make them more functional.
The game is well developed and playable. Our hero can direct the blows by moving the directional keys and combining them with the fire key. Graphically it is very nice and well animated. Minimalist but effective.
Summing up is a nice title to try, simple and playable.

by **Roberto Del Mar Pirazzini**



OUR FINAL SCORE

» Gameplay 75%
Once you redefine the keys it lets you play easily.

» Longevity 65%
It's not super long but it's fun.

OVER HORIZON

Year: 1991
Editor/Developer: HOT BB, PIXEL, TAKARA
Genre: Shoot'em up
Platform: Nintendo NES

It's fun to look back at Nintendo's library of shooters.
Despite the slow processor, it has an excellent library within the genre; of course, much of that is due to its market dominance.

I've always been fascinated by titles that made it to Europe but skipped the United States. Fascinated is not the right word, more like jealous. The NES version of Parodius is excellent and is one of my favorite shooters for the system. Another great game that received little attention is Over Horizon. This hidden gem is one of the best shooters on the 8 Bit Nintendo and has some really cool features.

Over Horizon keeps its weapon system simple and straightforward.

There are three main weapons: lasers, bombs and search bullets. These can be upgraded three times although the differences are not too dramatic. You can also pick up two options, a standard for the genre.

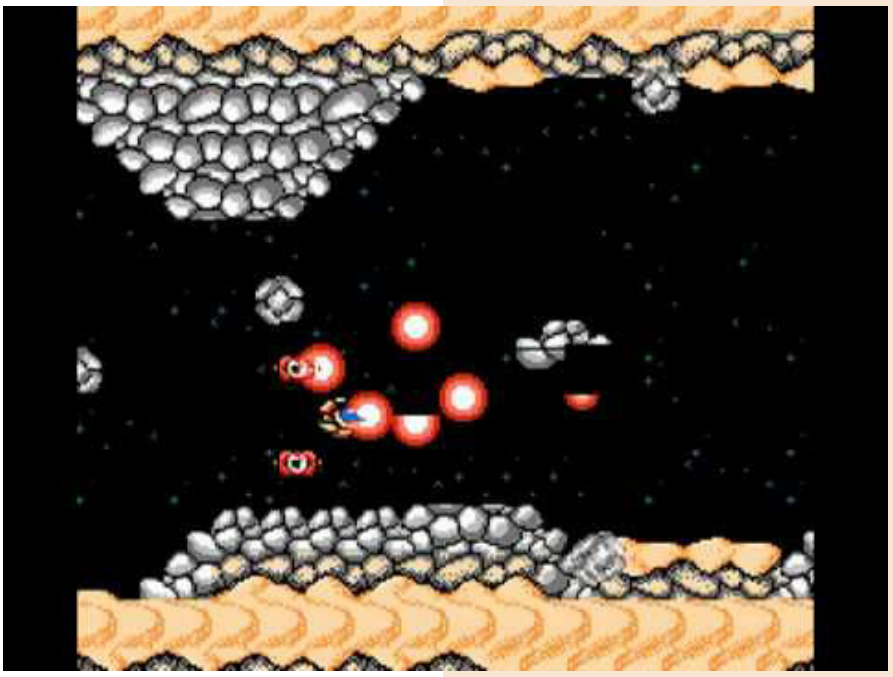
These options reflect your fire and can be slightly repositioned by pressing both buttons (A and B). An interesting feature is the ability to fire both forward and backward. This type of fire is essential to complete the game since several times we will be attacked from behind.

At first glance, the weapon system isn't very impressive. However, the game offers us the tools to customize it with its brilliant Edit mode.

This mode is one of the most innovative features in the genre, letting us customize each weapon with an intuitive interface.

The way it works is simple; we are given five points per weapon to give them the characteristics of the other two. The more points are given the more the weapon will favor the one you choose.





For example; if you want to modify the laser: dumping four points into the homing and one into the bombs will create a laser that creates a small explosion on impact. It gets really interesting if we try to strengthen the weakness of a weapon. The reference shot is weak, but adding laser points makes it deadly. The game also provides a trial level to test your modifications.

Besides the modification mode, the level design is also what makes Over





OUR FINAL SCORE

» Gameplay 95%
The weapon system modification is outstanding and innovative. Simple to play and a lot of fun.

» Longevity 80%
Challenging but not impossible. The experienced player will never be in serious difficulty. But it's a title that lets you replay often.





Horizon outstanding. Each world is characterized differently and presents unique challenges. Compared to most games in the genre, Over Horizon relies heavily on the setting to increase the challenge. For example, the fifth level features many dips in the water that slow movement and waterfalls that unexpectedly push us down. Third World ice blocks have to be pushed manually to create passages. The tricky part is that they bounce and there are even enemies that can throw them around! In all these innovations and special features, however, there is one flaw, the length! Despite the canonical six levels, the game does not present major difficulties and, after a few games, you will be able to complete the mission. It is also very generous with the respawn points. The only area that presents a real challenge is the sponge boss area. With the very high rate of extra lives present, however, most of the difficulties can be overcome without too many problems. With two more levels it could have been perfect.

Both artistically and technically Over Horizon is a marvel. The sprites are deliberately small to allow for the rich amount of detail in the background. The backgrounds are outstanding, from the giant plants and stems of the early stage to the ice streams of the third stage. Bosses are often huge mechanical contraptions that fill the screen and wouldn't look out of place in a 16-bit game. However, all of this visual splendor comes at a cost. There's a lot of slowdown and sprite flicker that impacts gameplay. But I think it's a worthy compromise.

In closing

Over Horizon is a fantastic game and should have made it to the US which didn't happen and in Europe it only made it to Germany.

Its modification mode is incredibly innovative and fun and gives the game replay value. A true masterpiece for the NES.

by **Roberto del Mar Pirazzini**



NEW GAME

LITTLE MEDUSA

Year: 2018
Developer: Mega Cat Studios
Genere: Puzzle Game
Platform: Super Nintendo, Sega Megadrive, Nintendo Nes
Vested version: Super Nintendo
Website: https://megacatstudios.com/collections/16-bit/products/little-medusa-snes

It has been centuries since the brutal Titans were sealed and banished, the Olympian gods have taken control over Mount Olympus.

The world of the gods is peaceful and they live together in harmony and even have children, nicknamed the Little Ones of Olympus.

We'll be playing as Artemiza, daughter of Zeus, who unfortunately due to the envy of the wicked Fiora, an evil minor goddess, has become a small jellyfish, unrecognizable to everyone. While all the gods are looking for us, the wicked Fiora frees the Titans and kidnaps all the Little Ones of Olympus. At this point it will be up to the little jellyfish to free the others from the clutches of the terrible monsters.

A classic story that is explained in the introduction of the game and creates the right atmosphere.
The game that Nith passed me is a nice product of the new generation of games for Super Nintendo developed by the team of Mega Cat Studios.
The graphics have a very fun style, somewhat reminiscent of the Zelda games due to the top-down perspective. The game itself uses themed levels, which bring out the color of the world we will be in.

The character models are well designed and take good advantage of the console. Unfortunately, they aren't very varied, which is a shame, since the game actually thrives on its looks. Monotonous music, but since it's actually a puzzle game at the end of the day it's for the best.











OUR FINAL SCORE

» Gameplay 80%
A nice puzzle game with action moments like opposing boss battles. I like it!

» Longevity 75%
Playing it in normal mode, the puzzles have an increasing difficulty and the boss fights are addicting. Impossible the Olympic version.



It's a pretty quiet background theme, though I would have preferred something more relaxing. Sound effects done correctly but minimal.

The game is a puzzle game with some action segments. From start to finish you'll have to turn enemy thunders into stone and use them as building blocks to reach certain parts of the levels. You can push them to the desired location or destroy them if they block your way. You should keep in mind that you can only push them in a straight line, so often you'll need patience and skill to guide them correctly to the desired position.

Enemies kill us after a touch. While the puzzle levels have an increasing difficulty, the boss battles make things more interesting.
To kill the boss you would have to use three petrified enemies, but doing so while it's in motion is no easy. However, I found this change of pace to be quite fun and enjoyable.
Hardcore players can try the "Olympic" mode, which is the Super Hard version of the game. Same levels, but without the possibility to continue: only the lives assigned at the beginning. This mode is only recommended to expert players.

Little Medusa is a tough-as-nails puzzle game. It's a fun game and a title that deserves to enter the library of games for the SNES.
You can purchase it from the Mega Cat Studios link provided in the description.

by **Ingrid Poggiali**





XENOCRISIS

Year: 2019
Editor/Devloper: Bitmap Bureau
Genre: Shoot em up Top Down
Platforms: Pc, Sega Megadrive, Sega Dreamcast, Neo Geo Aes/svs
Tested version: Sega Megadrive

Smash TV has been a part of most gamers' lives, whether it was their Saturday afternoon routine around arcades, seeing the box art in a magazine, or playing one of the many games inspired by the arena shooter genre it created.
Along with this classic, we all dreaded facing the iconic Xenomorph from the Alien series.

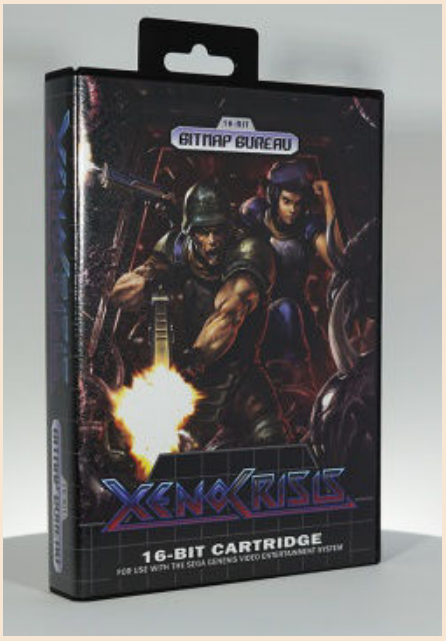


HR Geiger was certainly aware of the nightmares he would induce with his creation, but he had no idea of the inspiration it would ignite in millions of people around the world.
Bitmap Bureau have taken the iconic gameplay of Smash TV and sprinkled some classic slasher action with Xenomorphs, mixing them together in their newest 16-bit run 'n gun game, Xeno Crisis.

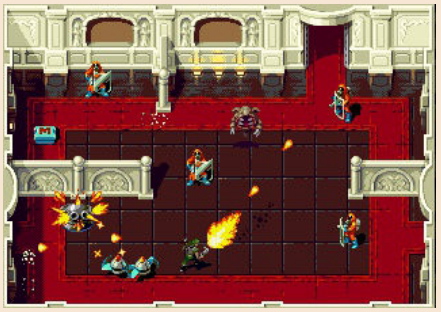
Xeno Crisis is a top-down arena shooter available for PC or physically on Genesis/Mega Drive, Dreamcast and Neo Geo AES/SVS (yes, that's right; Xeno Crisis will be on one of those giant cartridges that can double as a coffee table, ed.).

When we turn on the video game we'll be greeted by a movie made in beautiful pixel art by Henk Niebord, previously known for his work on Contra 4, Shantae and The Misadventure of Fink, just to name a few. The mini-film tells the story of a team of space marines sent to investigate a distress signal. Sound familiar? I hope so, and you'll recognize it best because you'll see nods to the entire 80's Sci-fi world, from Aliens and Terminator to Tremors and Predator.

Before landing on the hapless outpost, we'll select either a male or female marine.
The female marine is noticeably faster but frail, while the male marine can take more damage but is slower.



Before we get into the nitty-gritty, I'll tell you that there are two difficulty modes: Easy and Hard. Here guys do not mess around! There is no such thing as "normal" and the difference is obvious. Difficult mode makes the game a real hell for our marines and, until we are able to understand the







movements of the enemy, I recommend switching to the more "livable" Easy mode.

Xeno Crisis immediately reminded me that growing up with challenging games didn't make me any better. I became very familiar with the "Continue" screen, but it has quite a bit of personality, which helped alleviate the recurring failure I experienced.

When you choose to continue, the Marine punches an alien in the face and returns to battle with the temporary invincibility provided by an explosion! I have never experienced such a satisfying feeling on a "Continue" screen.

The action is fast, fluid and responsive, while the graphics are truly mind-blowing, especially considering the age of the console. The audio is great too, with an incredible soundtrack by Savaged Regime and some surprisingly clear voice samples (the Mega Drive was never very good at handling speech). Because it's essentially a dual-stick shooter, Xeno Crisis makes good use of the Mega Drive's six-button controller to give us a single button per direction (X is left, A is down, Y is up, and B is right). If you've played the SNES version of Smash TV, you'll be familiar with the system. Grenades are activated by the Z button, while the C button activates the dodge command, something you'll have to make use of in later levels when the action heats up. Playing with the standard 3-button controller is possible, but not as intuitive.

Buy it, master it, enjoy it. This game is one of the reasons we still play on our old consoles.

Neo Geo Version

We took a look at the Neo Geo version as well. There are no major technical differences, but certainly there is a greater fluidity of action and a more enveloping sound.

The control system instead suffers a lot of the only 4 keys, but with a couple of tricks you will get used to it.

The vote in this case is not changed.

by **Carlo N. Del Mar Pirazzini**

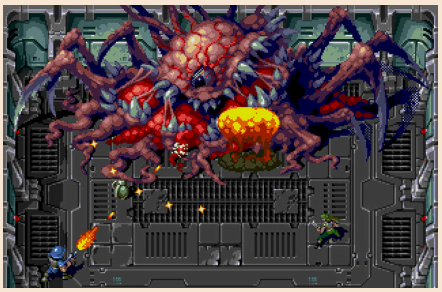
OUR FINAL SCORE

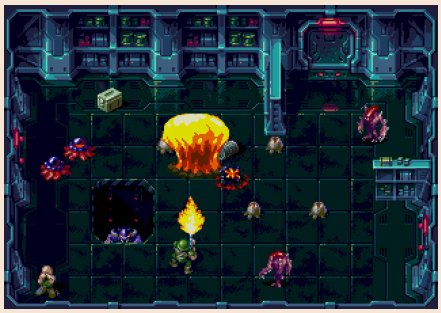
» Gameplay 90%

Control system not easy with a three-button pad, but amazing with the 6-button one. Playable and fast-paced. Shoot, destroy, collect weapons and... keep going!

» Longevity 98%

It's tough! It's difficult! It's a rude game that will punish you! But you will never stop playing it... I don't stop. Every time I turn on the Megadrive, I play with it.







BISHOUJO SENSHI SAILOR MOON R

Year: 1994
Developer: Angel Studios.
Genre: Action/Puzzle
Platform: Game Boy



I'm not a fan of Sailor Moon, but I really appreciate it, especially the anime and the merchandising related to it!

Among other things, the creator Naoko Takeuchi is married to the legendary Yoshihiro Togashi, creator of "Yu Yu Hakusho" and "Hunter X Hunter", after this brief (and perhaps useless info), I wanted this game from the first moment my eyes fell on it, why? Simple, it is a very unique game totally Japanese, and quite "varied", contains puzzles, also contains this "walkthrough" that if you do not know Japanese you wonder "what is it?", and then there is my favorite part: the adventurous one where you beat everyone.



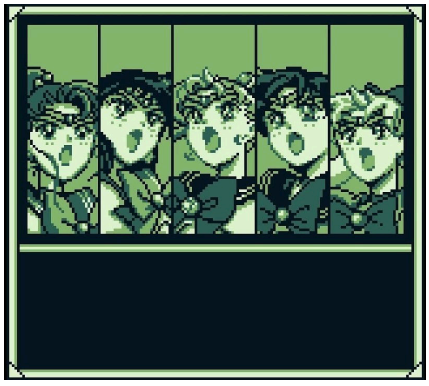
Perhaps it is not as long as you might hope, it took me about six minutes exactly to finish it with Sailor Mars. I forgot is unfailing arrival Mamoru (Martius) at the end of the missions, which pulls his useless pink as always.



Unbelievable.
I like it though, it's very 90's, and I would have loved to see Takahashi's Game Boy games come a little closer to this one... But no.

You want my opinion? Buy it if you like weird games, I liked it a lot (even though it doesn't look like it).

Either way. "I'm a champion of the law, I'm a fighter who dresses like a sailor!



I'm Sailor Moon! And I'm here to punish you in the name of the Moon!"

by **Barbara "Morgana" Murgida**







OUR FINAL SCORE

» Gameplay 80%
Simple action part, a little more complicated puzzle part. But fun.

» Longevity 50%
It ends in a short time. Too bad!

VASTAR

Year: 1983
Publisher: Sesame Japan Corp.
Genre: Mech Shooter
Platform: Zilog Z80 systems

Having played VASTAR belongs to the memories of my summer vacation in 1984. This decent horizontal scrolling mech shooter was the last game produced by Orca. The Japanese company declared bankruptcy at the same time the game was finished. The game was released starting in December 1983 under the alias Sesame Japan Corp.

Designed by Toshiaki Otha and his staff, it ran on Zilog Z80 systems at 3.072 Mhz with a 1.5 Mhz AY8910 sound chip. It admitted up to 2 players who alternated the following controls: an 8-way joystick and 2 buttons (laser and shield).

VASTAR looked like a cross between Gundam and a Zaku. Set in a post-apocalyptic future, the goal was to destroy enemies by moving from one base to another. Each time, VASTAR was catapulted out of a base to the sound of a distinctive jingle.

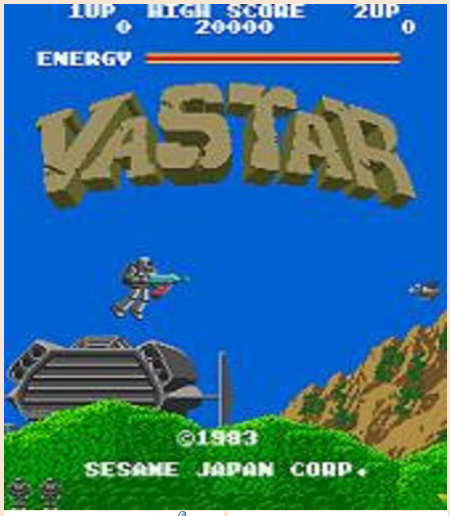
During the course of the game, VASTAR could transform into a fighter plane (the transformation was not controlled by the player). In addition, strange landscapes such as Mount Rushmore, the ruins of the Statue of Liberty, and the Moai on Easter Island could appear in the background.

VASTAR was not very easy. My brother and I, in order to play more with less money, had organized ourselves as follows: my brother maneuvered VASTAR while I took care of the laser and the shield (being careful about energy consumption).

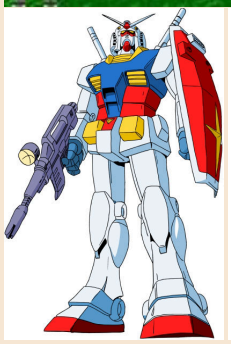
Searching on the Internet, you can find the ROM for the MAME emulator and tricks to overcome some levels. Other similar games are Formation-Z (1984) and Aeroboto (1985) but I prefer VASTAR and dream for a moment to own a Mobile Suit like Amuro Ray with his Gundam.

by **Alberto Apostolo**

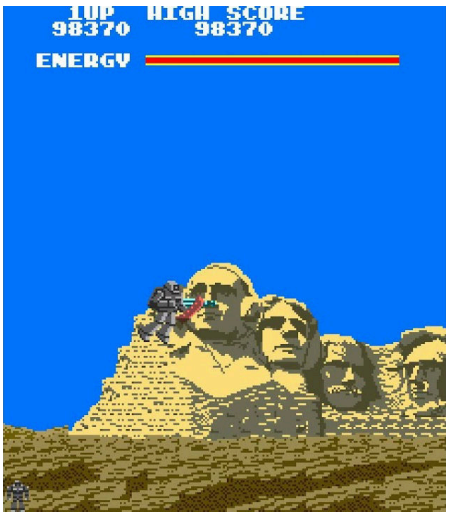
















OUR FINAL SCORE

» Gameplay 75%

Discreet mech shooter. Enemies and obstacles not to be underestimated, if you do not want to waste the "lives" obtained with bonuses.

» Longevity 75%

Every collector absolutely must have it to play it once in a while.

STRIDER

Year: 1990
Editor: US Gold
Genre: Action/Platform
Platform: Comnmodore 64

Also this year Christmas is approaching (when this article was written), fortunately without the same restrictions of last year but with streets and city centers lit up since mid-November; hoping it's a good sign after these almost two years quite grey under all points of view. So the day of my forty-fourth birthday has come nearer; I thank all the editorial staff that punctually wished me well.

Among the many games I was spoiled for choice, for this issue I put in the spotlight a game that was successful on other platforms but whose conversion on the breadbin did not give the same results, leaving the fans with a bitter taste in their mouths. Strider, platform game landed on consoles and, given the success, it was decided to convert it for home computers, with a conversion that left too much behind, especially for the reduced levels...

But let's talk about the purpose of the game: the setting is in a futuristic Russia, populated by machines, with the protagonist armed with a sword that leaves a huge circular trail with each slash. He must make his way through five levels full of soldiers, robots and bosses present in the middle of the level and also at the end. The gameplay, which in my opinion was not so bad is the classic platform game, where you have to jump here and there on platforms in an acrobatic manner and avoid precipices. Falling involves the direct loss of a life even if the energy bar is full.

The sound is made only of the blows given and received, so music is totally absent on the Commodore 64 conversion.

In terms of longevity could have lasted months if only there were the levels, a flaw that I felt more than the missing of some of the backgrounds.
Despite these flaws that have outweighed the merits, I still decided to play and complete it.
My advice is not to underestimate too much this conversion and enjoy it as far as it has to give, and then play it on other platforms where it has been converted with better results.

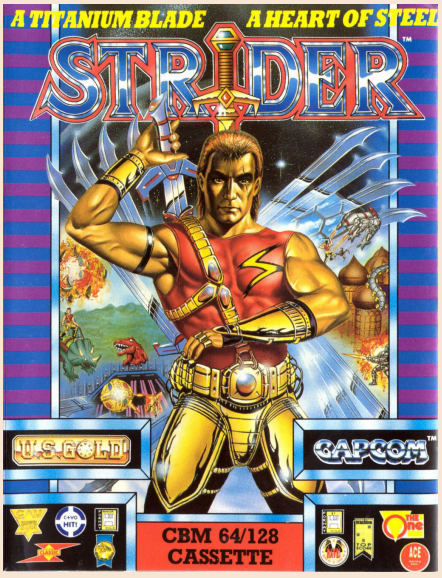
Speaking instead of levels, they will seem quite challenging at first, when you do not know the path or the location of enemies, but after a few games will undoubtedly be easier and smooth. Despite everything, even the second chapter was converted for home computers and without spoiling anything, let alone spoil the expectations for those who had never played it, there were no improvements hoped for!

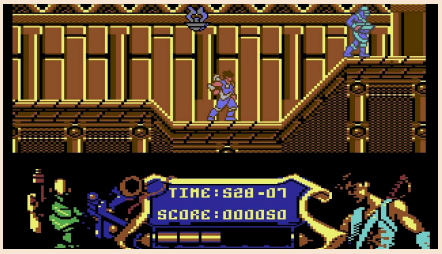
However, a big game like Strider converted for home computer, hardly gives the same results of a console. At least at the time it was coded, differently by what happened a few days ago with the unthinkable porting of Sonic the Hedgeog. A masterpiece that has left everyone speechless and a great Christmas gift!

Who knows, maybe Strider in the future will be converted again to give it the value it deserves, thanks to some nostalgic and willing programmers as happened with other titles. Besides being a collaborative family, retrogaming is almost becoming a religion.

Happy holidays and a wonderful beginning of 2022 to you all.

by **Daniele Brahimi**







OUR FINAL SCORE

» Gameplay 75%
Sufficient controls with few inaccuracies.

» Longevity 80%
Reduced levels, but compensated by the difficulty.

Taking pleasure, if it were up to "us", it'd be a duty for everyone.

*(freely taken from a sentence by **Tinto Brass**)*

On December 19, 2021, the free release of **Sonic the Hedgehog** for **Commodore 64/128 + REU** (PAL and NTSC) was announced. A masterpiece made possible on the C64 only thanks to a hardware extension.

The author, known in the C64 community with the nickname Mr. SID, says that it is the first game made exclusively for REU with about 64Kb of code hand-translated from the Z80 architecture to the 6502 one (more details can be easily found searching the Net).

But that's not the point. On social media, animated discussions have found space in the post between factions of purists of retrocomputing and those who do not disdain to use emulators or other systems.

It's not the first time (and it won't be the last) that RetroMagazine World will reaffirm its openness to all those who share the passion and the desire to have fun with retrocomputing.

Of course, it's a great feeling to play with an original device that's still in great condition (considering that anything will tend to deteriorate over time until it no longer works). However, it is our opinion that using emulators does not break any moral rules. In fact, almost all electronic systems are designed from an emulator. This method, allows teams to reduce the time of realization and (above all) not to leave errors carved in the silicon of a chip or fry a board at the time of power-on.

RetroMagazine World invites all enthusiasts to fully enjoy all the tools available (real and vintage hardware, software emulators, FPGA board reproducers...) and will always be happy to give space to those who propose interesting things based on any platform, because in the end what really matters is to have fun together and enjoy these gifts that are often given to us for free.

Just before going to press, we hear that one of the most gifted programmers of the Commodore 64, the same **Sarah Jane Avory** who gave us a masterpiece like **Briley Witch Chronicles**, is trying to use the REU to create games. Now that's exactly the positive effect we wanted to see from the Sonic opener. Let's stay tuned, we might see some good stuff!

Alberto Apostolo & **Francesco Fiorentini**

Disclaimer

RetroMagazine World as an aperiodic magazine entirely ad-free is a non-profit project and falls off any commercial circuit. All the published material is produced by the respective authors and published thanks to their authorization.

RetroMagazine World is licensed under the terms of: **Attribution-NonCommercial-ShareAlike 4.0 International** (CC BY-NC-SA 4.0) https://creativecommons.org/licenses/by-nc-sa/4.0/

This is a human-readable summary of (and not a substitute for) the license. You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material

The licensor cannot revoke these freedoms as long as you follow the license terms. Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial — You may not use the material for commercial purposes.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

RetroMagazine World-Eng
Year 3 - Issue 12 - JANUARY 2022

Chief Editor
Francesco Fiorentini
Managing Editor
David La Monaca
Editing Manager
Marco Pistorio
Web Management
Flavio Soldani/Giorgio Balestrieri