



# RetroMagazine

World



future days are back

TANTISSIME

RECENSIONI



## GREEN BERET

AMIGA

COLECOVISION



**SPECIALE FAMILY BASIC: mettere un "computer" nel FAMICOM**  
**Software: RETRO FIGHTER ENGINE per C64 - ATARI ST Basic: AES e VDI -**  
**RETRO PROGRAMMING ITALIA: Come creare un gioco in BASIC su C64 (parte 2)**  
**Rubriche: WAR and VIDEOGAMES - AMIGA: Le 10 (+1) game intro spettacolari**  
**GAME BOY ADVANCE: le (S)protezioni - Add-on non ufficiali per ZX Spectrum**  
**... rubriche, interviste e molto altro ancora!**

## Abbandonate la zona di comfort!

Che cos'è la zona di comfort? WikiPedia definisce la zona di comfort come *...uno stato psicologico in cui un individuo percepisce tutto come familiare e si sente a suo agio...*

Provando a trasportare questa definizione nel mondo del retrocomputing, possiamo immediatamente associare la zona di comfort alle macchine, ai sistemi o ai giochi con cui abbiamo familiarità. Probabilmente ci verranno subito alla mente i sistemi che abbiamo utilizzato da ragazzini e/o con cui siamo cresciuti e che hanno contribuito a formare la nostra esperienza di giocatori, programmatori o semplici appassionati.

Ma cosa succederebbe se decidessimo di abbandonare questa zona di comfort? Cosa succederebbe se decidessimo di provare ad utilizzare e a conoscere in maniera più approfondita i sistemi a noi non familiari? Sicuramente all'inizio pagheremmo lo scotto della mancanza di conoscenza specifica. I comandi che siamo abituati ad utilizzare in modo quasi meccanico e le operazioni che facciamo abitualmente senza nemmeno pensarci, probabilmente non avrebbero l'effetto sperato e proveremmo un senso di frustrazione. Ed è proprio in questo momento, quando il nostro istinto vorrebbe farci abbandonare tutto per tornare a contesti a noi più familiari, che invece dobbiamo insistere ed allontanarci dagli assodati schemi mentali.

Ebbene, se saremo in grado di farlo, si spalancheranno di fronte a noi mondi inesplorati, dove la gioia della scoperta potrebbe farci tornare quei ragazzi che eravamo 30/40 anni fa. Tutti noi quando abbiamo preso in mano il nostro primo computer, abbiamo provato una sorta di timore reverenziale verso un oggetto sconosciuto. Abbiamo dovuto leggerne il manuale o chiedere aiuto ad amici più esperti per caricare un gioco, scrivere un programma o anche soltanto per connettere cavi e periferiche.

Quella stessa sensazione che ho provato quando ho deciso di mettermi in gioco con macchine che non conoscevo affatto. Tutte le mie certezze sono svanite in un attimo, soltanto per visualizzare il contenuto di un disco o di una cassetta. Anzi, per alcuni sistemi, le mie certezze hanno cominciato a vacillare di fronte alle possibili configurazioni ed alle opzioni presenti nell'emulatore... Per fortuna oggi abbiamo a disposizione migliaia di risorse dalle quali attingere per familiarizzare con questi mondi a noi sconosciuti.

Cosa aspettate quindi? Uscite dalla vostra zona di comfort, qualunque essa sia, e provate ad utilizzare sistemi che ancora non conoscete. Un nuovo mondo vi sta aspettando. Un mondo per voi inesplorato, ma con quella ricchezza di colori che ha già affascinato migliaia di utenti, con quei giochi che hanno fatto impazzire generazioni e con quei comandi che hanno tenuto occupati programmatori per intere notti!

Fidatevi, abbandonate la vostra zona di comfort e ne vedrete delle belle!

**Francesco Fiorentini**

## SOMMARIO

◇ Family BASIC - un "computer" nel Famicom	Pag. 3
◇ GameBoy Advance - le (s)protezioni	Pag. 8
◇ Add-on non ufficiali per Sinclair ZX Spectrum	Pag. 10
◇ ColecoVision	Pag. 14
◇ RetroFighter engine per Commodore 64	Pag. 19
◇ Atari ST Basic - AES e VDI	Pag. 22
◇ Come creare un gioco in BASIC per C64 - pt. 2	Pag. 26
◇ C128 in modalità 80 colonne - parte 3	Pag. 32
◇ Frattali in BASIC in salsa 8-bit - Parte 4	Pag. 36
◇ Intervista ad Andrea Contato	Pag. 40
◇ Diamo un'occhiata a Lykia - The Lost Island	Pag. 44
◇ War and Videogames	Pag. 46
◇ Amiga: le 10 (+1) game intro più spettacolari	Pag. 51
◇ Talent Scout	Pag. 57
◇ Randoom (C64)	Pag. 58
◇ Randominer (C64)	Pag. 59
◇ Green Beret (Amiga)	Pag. 60
◇ Metal Dragon (MegaDrive)	Pag. 62
◇ Star Ocean (SNES)	Pag. 64
◇ Attack of the Petscii Robots (C64)	Pag. 66
◇ Klonoa 2 Lunatea's Veil (PS2)	Pag. 68
◇ Twinkle Tale (MegaDrive)	Pag. 70
◇ Zaku (Atari Lynx)	Pag. 72
◇ S.O.L.O. (MSX)	Pag. 74
◇ Rogue 64 (C64)	Pag. 75
◇ Sigil (PC)	Pag. 76
◇ Witch n' Wiz (NES)	Pag. 78
◇ Doraemon (GameBoy)	Pag. 79
◇ Ski (C64)	Pag. 80
◇ Crack Down (Arcade)	Pag. 81

### Hanno collaborato alla stesura di questo numero di RetroMagazine World (in ordine sparso):

- Alberto Apostolo
- Dr. Andrea Q.
- Carlo N. Del Mar Pirazzini
- Daniele Brahimi
- Mic the Biker Novarina
- Francesco Fiorentini
- David La Monaca
- Leonardo Miliani
- Barbara "Morgana" Murgida
- Takahiro Yoshioka
- Roberto Del Mar Pirazzini
- Ingrid Poggiali
- Gianluca Girelli
- Giampaolo Moraschi
- Querino Ialongo
- Gianluca Alberico
- Felice Nardella
- Beppe Rinella
- Alessandro Albano
- Immagine di copertina: **Giuseppe Mangini**
- Layout di copertina: **Carlo N. Del Mar Pirazzini**





# Family BASIC - Mettere un "computer" nel Famicom

di Takahiro Yoshioka (trad. Carlo Nithaiah Del Mar Pirazzini)

Quando Nintendo ha rilasciato la sua prima console, il Famicom, il nome completo della console era Nintendo Family Computer.

All'inizio degli anni '80 il confine tra console per videogiochi e computer domestici era confuso.

Alcune console, come l'Odyssey 2, avevano una tastiera completa (a membrana), ma erano più simili a console per videogiochi.

Alcuni computer come il Commodore 64 avevano una tastiera completa e un'unità disco, ma potevano essere ricchissimi di giochi.

Al lancio di Famicom nel 1983, gli unici software disponibili per la macchina erano i port giochi arcade.

Entro l'anno successivo (1984) Nintendo aveva rilasciato un nuovo prodotto per Family Computer che aveva lo scopo di fare di più che giocare a un altro videogioco.

Questo prodotto è stato rilasciato come Family BASIC il 21 luglio 1984, ma è stato rilasciato solo in Giappone.

Il pacchetto Family BASIC includeva due componenti hardware. Il primo la cartuccia Family BASIC, il secondo la Family Keyboard.



Fig. 1 - Immagine da Wikipedia

La cartuccia Family BASIC era unica al suo tempo. Conteneva 32 KB di Rom di programma (PRG) e 8KB di Rom di caratteri (CHR-).

In una seconda versione venne fornita di 2KB o di 4KB di Ram di lavoro (W).

All'interno era presente un chip logico aggiuntivo per disabilitare il chip PRG-ROM durante la lettura o la scrittura in W-RAM poiché entrambe le memorie si trovavano sull'indirizzo della CPU e sui Bus dati.

Altre due caratteristiche della cartuccia erano un interruttore on-off di backup del software nella batteria. Quando l'interruttore era acceso, il contenuto in W-RAM non veniva



Fig. 2 - Family Basic - Immagine da bootgod

perso allo spegnimento del Famicom.

La seconda caratteristica unica è che la batteria interna era totalmente sostituibile dall'utente perché in formato AA. Erano quattro le versioni del software Family BASIC rilasciate. Le prime tre, v.1.0, v.2.0A e v.2.1A, avevano lo stesso aspetto esternamente, contenevano 2KB di W-RAM ed erano vendute anche con la Family Keyboard. Esisteva anche una versione per Sharp My Computer C-1 Tv chiamata Playbox BASIC.

La versione finale v.3.0 è stata rilasciata nel 1985 con 4KB di W-RAM e venduta separatamente dal pacchetto Family BASIC e dalla tastiera.

La tastiera presentava 72 tasti ed era a struttura meccanica ma con la grandezza dei pulsanti decisamente più piccola rispetto alle classiche tastiere. Somigliava per aspetto a quella del C64 e a quella per lo Sharp X-1 e si collegava al Famicom attraverso la porta di espansione a 15 pin. Era realizzata in modo da avere una matrice di due "colonne" e otto "righe". Ogni tasto era unico e con funzione singola.

Era provvista anche di un paio di jack audio mono da 3,5 mm etichettati come "write" e "red" nelle prime versioni e successivamente come "Save" e "Load" nelle versioni





Fig. 3 - Family Basic V3 - Immagine da bootgod

post 1984. A questi jack era possibile attaccare un registratore di dati, acquistabile separatamente, che permetteva di registrare il lavoro svolto.

### IL BASIC

La famiglia BASIC era un port di HU-BASIC di Hudson Soft, assemblato in modo incrociato sulla linea di computer di Sharp. HU-BASIC, come la maggior parte dei BASIC per computer domestici, era basato su Microsoft BASIC. Molti comandi e convenzioni familiari per gli utenti che hanno familiarità con un dialetto del BASIC discendente da Microsoft. Il programma HELLO WORLD standard viene inserito ed eseguito nel modo consueto. LIST, NEW, CLS, SAVE e LOAD funzionano come previsto. (Puoi usare

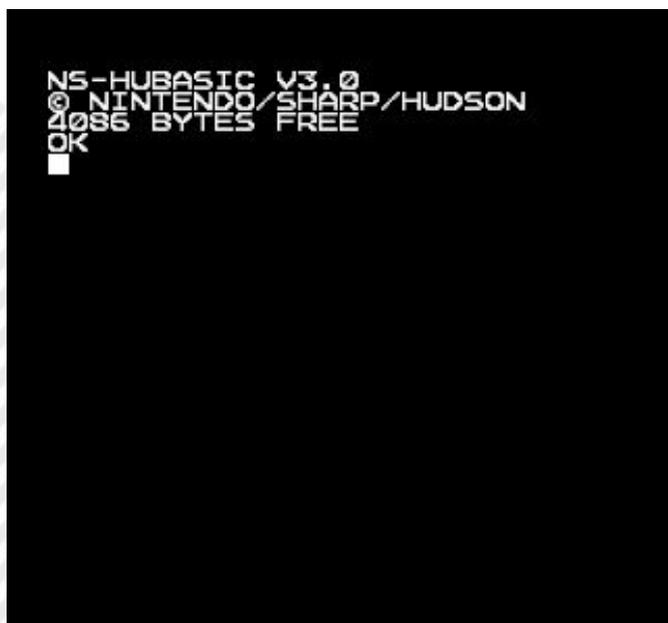


Fig. 4 - Il Basic

Maiusc + Clr/Home per cancellare lo schermo).

Ci sono comandi e differenze singolari in questo dialetto del BASIC. A differenza di un Microsoft BASIC, Family BASIC funziona solo con numeri interi, non decimali o virgola mobile. Se si digita STAMP 3/2 e si preme Invio, il risultato è 1, non 1,5. Il testo in inglese minuscolo non è presente, ma il tasto kana ti consentirà di digitare katakana o inglese carattere per carattere.

C'è un comando SPRITE per definire e visualizzare gli sprite, un comando MOVE per spostare gli sprite, un comando PLAY per riprodurre musica, un comando PALET per cambiare le tavolozze degli sprite o lo sfondo/testo e i comandi STRIG e STICK per prendere input dagli sprite azione e direzionali del gamepad.

Ci sono due importanti comandi LOAD e SAVE in Family BASIC. I normali comandi LOAD e SAVE caricano un programma da nastro o salvano un programma su nastro. La grafica di sfondo viene caricata nei 2KiB interni di RAM video all'interno del Famicom. Utilizzare i comandi LOADS e SAVES per caricare o salvare dati grafici in background da/su nastro per Family BASIC v3.

Il pacchetto Family BASIC includeva un ampio manuale per aiutare a insegnare alle persone come programmare in BASIC. Aveva anche un foglio che mostrava tutta la grafica disponibile per sfondi e sprite. Ovviamente i manuali sono in giapponese così come gran parte del testo visualizzato nel programma stesso.

E' possibile trovare le scansioni dei manuali e diverse traduzioni a questo indirizzo web: <https://www.famicomworld.com/forum/index.php?topic=11959.0>

### FAMILY BASIC v.1.0 - v.2.1

Un sistema molto "amichevole" per il tempo. All'accensione appare una schermata "Computer", una schermata "operatore" e da F1 ad F4 in basso (Fig. 5).

Premendo un tasto qualsiasi il computer inizierà a spiegare all'utente cosa è capace di fare. Dopo tre schermate di testo verrà chiesta la registrazione del nome.

Dopo una schermata di testo dove viene spiegato il menù successivo verrà chiesto se si desidera avviare "GAME BASIC". Premendo F1 (Si) e poi invio.

Questo porterà a un menù dove si potrà decidere di accedere al BASIC o al programma BG Graphic. Se si tiene premuto il tasto T all'avvio del Famicom è possibile ignorare



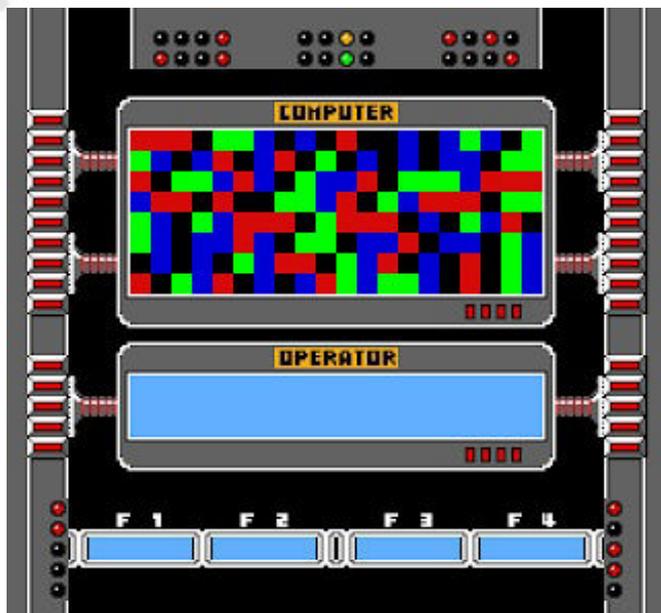


Fig. 5 - Schermata Operatore e Computer

la schermata iniziale e accedere direttamente al menù BASIC/BG (Fig.6).

Il pulsante immediatamente sotto Ritorno è il pulsante Kana ( ). Questo cambia la tastiera tra la modalità alfanumerica e la modalità Katakana. In questa modalità, solo Katakana è disponibile dai tasti alfanumerici. Se si vogliono numeri o simboli o segni di punteggiatura (tranne quelli usati in giapponese), si deve tornare alla modalità alfanumerica.

Il tasto GRPH ti darà la versione del personaggio. Il tasto Maiusc viene utilizzato per condividere i kana con i tasti YUIOP, fornendo la versione di quei caratteri e la versione più piccola del personaggio se mostrata sulla tastiera. Se il cursore è una casella lampeggiante, la tastiera è in modalità Katakana. Se è un quadrato lampeggiante, allora è in modalità alfanumerica.



Fig. 6 - Il menù Basic/BG Grpahic

Il programma BG Graphic consente di utilizzare i tasti di navigazione per visualizzare i riquadri sullo schermo. Utilizzando i tasti freccia su/giù/sinistra/destra potremo posizionare il cursore sul riquadro che si desidera modificare e premere la barra spaziatrice per impostare il riquadro.

Il Family BASIC contiene anche altri tre programmi, il Calculator, il music board e la message board. La calcolatrice consente di risolvere le equazioni più facilmente dal prompt di BASIC, la music board permette di comporre musica e la lavagna ci permette di aggiungere una serie di annotazioni. I dati di questi programmi sono salvabili sulla W-RAM ma non su nastro.



Fig. 7 - Il programma Calculator



Fig. 8 - Il programma Message Board

Per accedere a questi programmi è necessario all'avvio premere il tasto F2 e selezionare quale applicazione si vuole utilizzare.





Fig. 9 - Il programma Music Board

### Family BASIC V3

Questa versione migliora la navigazione interna dell'applicativo e permette di accedere in modo più veloce ad ogni singolo programma.

Sono presenti anche 4 minigiochi integrati che possono essere modificati e sono: Heart, Penpen Maze, Mario World e Star Killer.

Sono state aggiunte numerose opzioni nel BGTOOL e nel tool grafico. È stato aggiunto un comando Crash per rilevare le collisioni tra oggetti.

### L'eredità del Family BASIC

In Giappone, il costo di un "vero" computer era molto alto, anche per macchine a 8 bit come l'X-1 di Sharp o il PC-8801 di NEC o l'FM-7 di Fujitsu. La maggior parte

delle famiglie giapponesi non era pronta per un tuffo di migliaia di dollari per portare con sé l'ufficio a casa. Le console per videogiochi erano dispositivi molto, ma molto più economici e potevano lavorare meglio della maggior parte dei computer domestici. Il prezzo del Family BASIC era più costoso di una cartuccia per videogiochi standard (¥ 14.800, stesso prezzo del Famicom) ma inferiore a ciò che il mercato "professional" presentava. Family BASIC era uno strumento per aiutare i bambini ad imparare divertendosi.

È stato uno dei pochi programmi pedagogici pubblicati da Nintendo, insieme a Donkey Kong Jr. Math e Popeye's English Lesson, durante i primi anni del Famicom. Nei manuali Nintendo ha dovuto documentare come funzionava il Famicom in misura maggiore di quanto altrimenti avrebbe dovuto spiegare. Ciò ha aiutato alcuni programmatori come Satoshi Tajiri, il creatore di Pokemon, a iniziare permettendogli di esplorare l'hardware e utilizzare i concetti appresi per aiutare a programmare il suo primo gioco, Quinty.

Family BASIC, sebbene un modesto successo, non ha portato a una sorgente di interesse per la programmazione nel Famicom.

Il Famicom, nonostante il nome, era una console per videogiochi, non un computer di casa. L'archiviazione su cassetta era qualcosa che gli Stati Uniti si erano lasciati alle spalle negli anni '70 con poca nostalgia. Mentre il Family BASIC raggiungeva il massimo a 4 KiB nel 1985, nel 1977 le configurazioni minime dell'Apple II, del Commodore PET e del TRS-80 arrivavano a 4 KiB.

Il Family Computer per poter competere col mercato necessitava di un'unità a disco e, nonostante la realizzazione

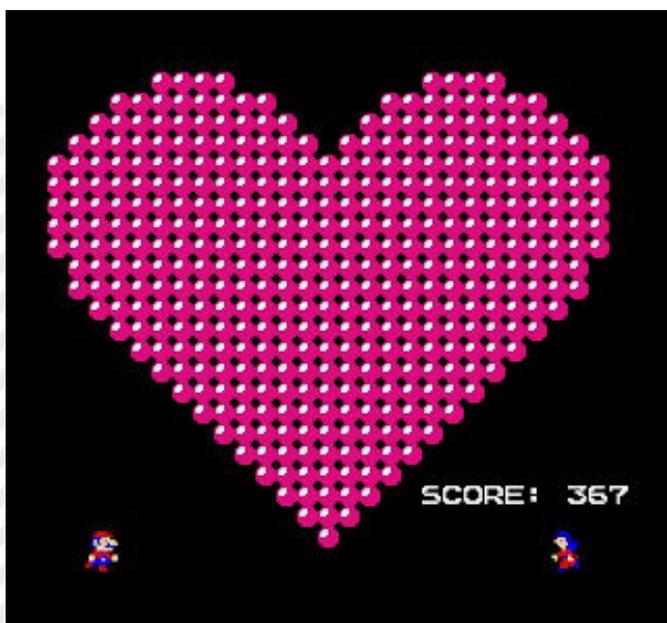


Fig. 10 - Heart

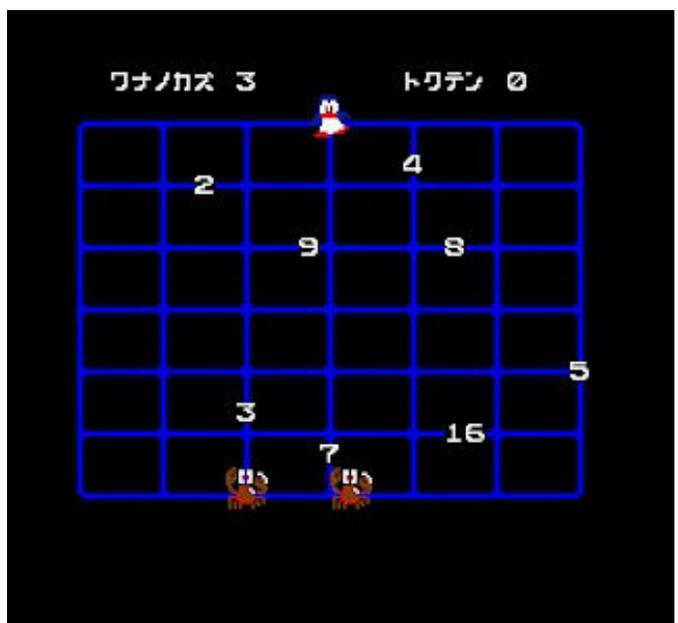


Fig. 11 - Penpen Maze



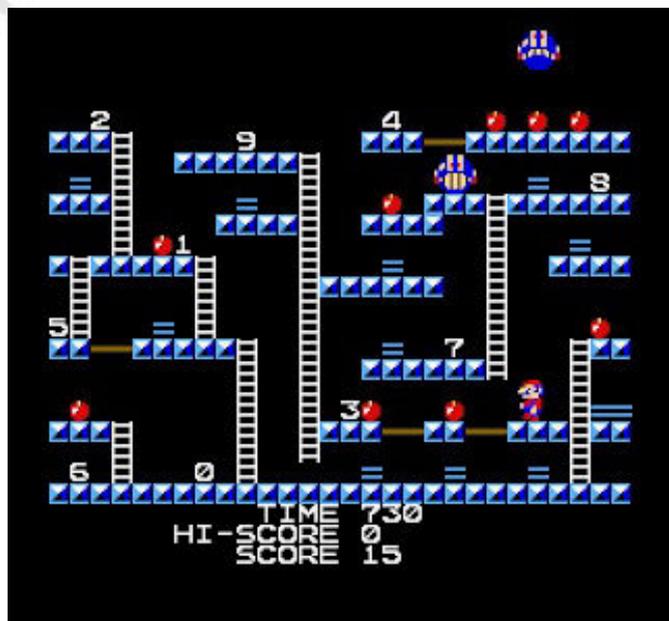


Fig. 12 - Mario World



Fig. 13 - Star Killer

del Famicom Disk System, non venne mai rilasciato alcun tipo di software per la programmazione. Quando Nintendo stava pianificando il lancio dell'uscita negli Stati Uniti del Famicom, aveva pensato di aggiungere anche il Family Basic nel progetto chiamandolo Advanced Video System. Nel prototipo erano presenti anche una tastiera e un registratore di cassette dati. Il prototipo della tastiera ha 72 tasti con i simboli katakana rimossi e il tasto kana trasformato in un tasto Caps Lock. Presumibilmente se il software Family BASIC fosse stato tradotto in inglese, avrebbe avuto la capacità di visualizzare il testo inglese minuscolo. Non è stato rilasciato e nulla è stato rilasciato per il NES ufficialmente o ufficiosamente durante la sua vita. Se Nintendo avesse rilasciato la tastiera per NES, la

tastiera avrebbe dovuto funzionare in modo molto diverso. Il NES ha meno porte di input e output tramite le sue porte controller rispetto a Famicom tramite le sue porte di espansione. Inoltre, le porte del controller devono condividere le linee di input con i controller. Con computer economici a 8 bit come il Commodore 64 ancora in circolazione alla fine degli anni '80, non aveva molto senso introdurre una cartuccia di programmazione.

#### Bibliografia

- Famicom World: <https://famicomworld.com/>
- Famibe No Yosshin (May 1993). "ZACNER II"
- Szczepaniak, John (August 2012). "A basic history of BASIC on its 50th birthday"
- Kojima, Ikuo (1994). Pseudorabies Publishing. p. 74.



Fig. 14 - Il prototipo dellAdvanced Video System





# GameBoy Advance - le (s)protezioni

di Dr. Andrea Q. - [www.retrofixer.it](http://www.retrofixer.it)

Canale youtube: <https://www.youtube.com/channel/UCEw0CQ8LKyA9jVvWXkEwp4Q>

Il Game Boy Advance (GBA), product code AGB-001, vide la luce del pubblico per la prima volta nel 2001 ed ebbe come suo predecessore un mai uscito in commercio "Project Atlantis", forse successore del Game Boy Color (nella foto grande sottostante potete vederlo sulla sinistra in una delle sue rarissime apparizioni accanto ad un DSi).

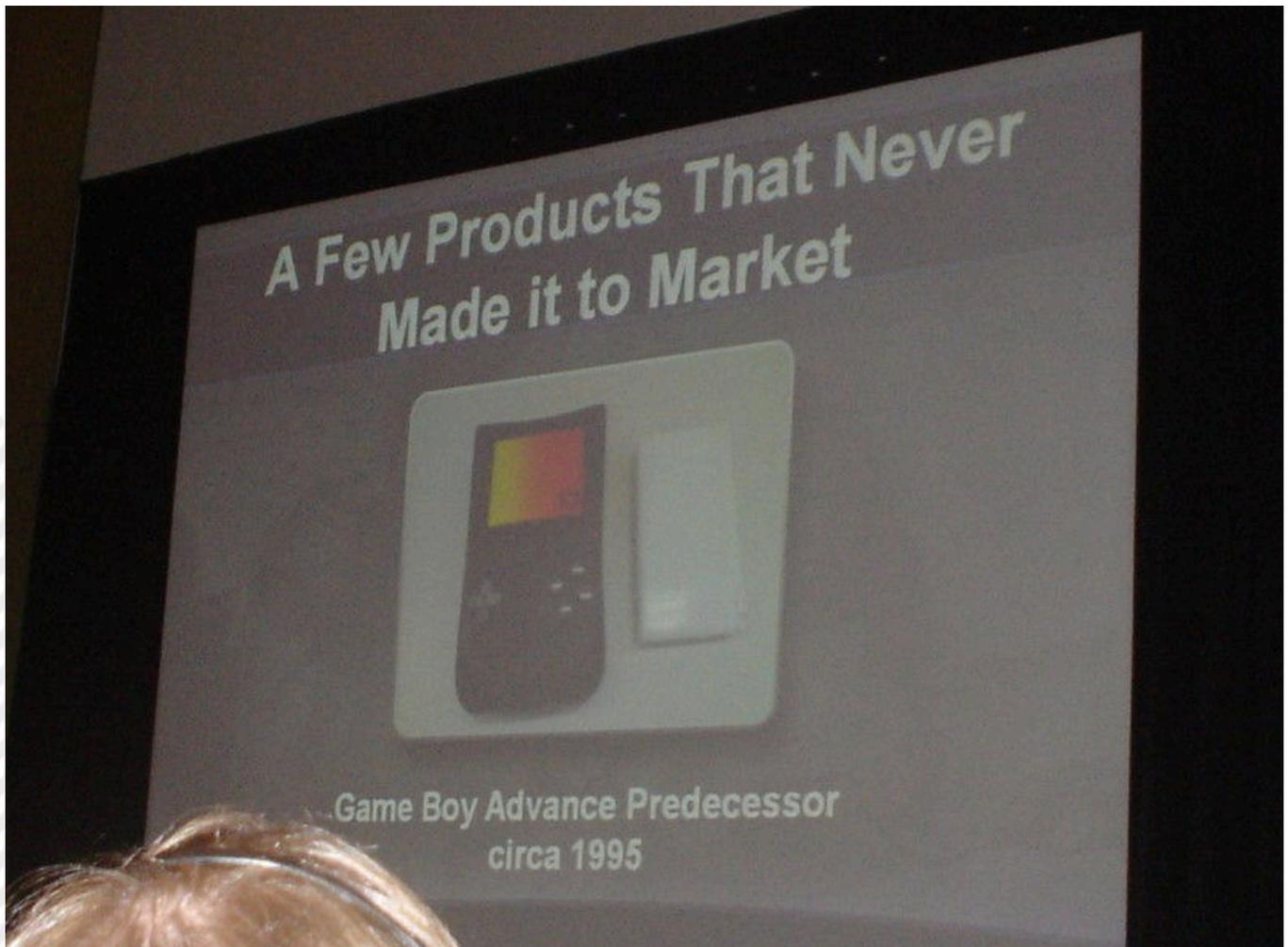
## PROTEZIONI

La console era region free e non disponeva di particolari sistemi di protezione (se non il solito checking del logo sia nella ROM interna del BIOS [offset \$3290] che nella cartuccia [offset 0x0004, per 156 bytes – controllo effettuato byte per byte] ma fu la prima, se non consideriamo la periferica aggiuntiva 64DD per N64, a possedere un suo sistema operativo. Una sorta di mini-BIOS che conteneva, in soli 16KB, il logo e poteva eseguire procedure



per una copia rapida della memoria, decompressioni, alcune operazioni matematiche di base, alcune funzionalità audio (utilizzate probabilmente da 3 giochi soltanto) ed una manciata di interazioni hardware a basso livello.

Le cartucce della serie "CLASSIC NES SERIES" (la versione fisica dei futuri titoli "Virtual Console") sembrano contenere particolari funzioni che provocherebbero problemi di





compatibilità se utilizzate con hardware Game Boy Advance non originale, in particolare le funzioni:

- CPU pipeline (selfmodifying code that shall NOT affect prefetched opcodes)
- STMDA write to I/O ports (writes in INCREASING order, not DECREASING order)
- SRAM detection (refuses to run if SRAM exists; the games do contain EEPROM)
- ROM mirrors (instead of the usual increasing numbers in unused ROM area)
- RAM mirrors (eg. main RAM accessed at 2F00000h instead of 2000000h)

Come nelle precedenti consoles portatili, anche nel GBA, il BIOS non può essere letto; infatti le funzioni di copia o di accesso alla memoria dello stesso sono controllate affinché non possano accedere all'area dove è memorizzato... Tutte tranne una, il software interrupt \$1F, anche chiamato MidiKey2Freq, utilizzato per convertire le note musicali da MIDI ad una frequenza udibile dall'orecchio umano.

Questo fatto non sfuggì al coder Dark Fader che già nel 2001 lo sfruttò per ottenere, in modo piuttosto lento e, byte dopo byte, il dump completo del BIOS su flash cartridge. Questo sistema, fino a poco tempo fa, rappresentava l'unico modo per effettuare il dump finché più recentemente, nel 2016, il dev endrift riuscì ad utilizzare una tecnica ROP per raggiungere il medesimo scopo.

### Che cos'è una tecnica ROP?

Rendendo le cose semplici possiamo dire che, con una programmazione oculata, le catene ROP possono aggirare il fatto che la memoria non sia "eseguibile". Più in dettaglio: le CPU recenti non permettono di eseguire in memoria codice arbitrario: ciò significa che se vuoi puoi copiare dati in memoria ma questo non significa che questi dati possano essere anche eseguiti; per farlo è necessario copiare tali dati in aree specifiche di memoria definite "executable" che sono impostate come tali dalla Memory Management Unit MMU (hardware che può essere separato o integrato nella CPU). Le tecniche ROP bypassano questo "blocco" selezionando con precisione la fine di diverse

funzioni che manipolano lo stato del sistema in modi specifici ed incatenano la fine di una funzione call con un jump alla fine di un'altra funzione che compie un altro specifico set di operazioni.

Con il suo nuovo metodo di dump (anche esso dumpa il BIOS direttamente nella SRAM di una flash cartridge) endrift ci dimostra come non sia sempre necessaria una vulnerabilità software per exploitare (sfruttare) un sistema perché i difetti hardware arrivano molto più in profondità.

Vista la mancanza di significative protezioni e la relativa facilità con la quale poter replicare una cartuccia GBA, i cloni dei giochi di questa console erano abbastanza diffusi all'epoca, soprattutto in Asia.

Questo articolo è stato breve e poco intenso ma il prossimo compenserà ampiamente. Arriverà infatti una trattazione sulle protezioni del Game Cube!

### ATTENZIONE: Esonero di responsabilità

Le informazioni contenute nel presente articolo sono riportate a solo scopo divulgativo. Non si garantisce che la presente documentazione sia priva di errori. Qualora queste informazioni vengano utilizzate per modifiche all'hardware dell'utente, è responsabilità dello stesso adottare tutte le necessarie misure di emergenza, backup, ridondanza e di altro tipo per garantirne la massima sicurezza di utilizzo. RetroMagazine World declina ogni responsabilità per eventuali danni causati dall'uso delle informazioni riportate nell'articolo.





# Add-on non ufficiali per Sinclair ZX Spectrum

di Alberto Apostolo

Negli anni '80 del XX secolo, diverse aziende producevano schede di espansione che consentivano di potenziare il Sinclair ZX Spectrum, salvare dati e programmi su floppy disk, stampare grafici e perfino far "parlare" il computer!

In questo articolo ho voluto descrivere quello che ha colpito di più la mia attenzione.

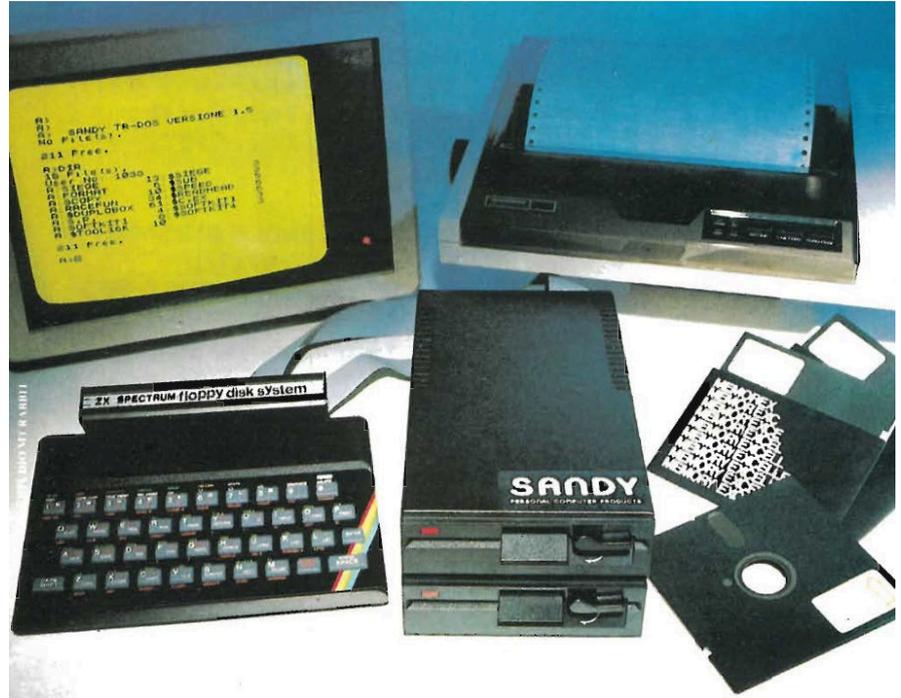


Fig.1

### SANDY

Sandy era una azienda italiana con sede a Senago, un piccolo centro abitato a nord di Milano.

Negli anni '80 del XX secolo mise in commercio una serie interfacce per floppy disk drive da 5.25 pollici (da 100 Kb, 200 Kb, 400 Kb) e da 3.5 pollici (da 200 Kb, 400 Kb). Dopo il collegamento e l'accensione del computer, il controllo passava al sistema operativo "SP DOS" (fig. 2). Per accedere ai contenuti dei dischetti era obbligatorio fornire una password.

Per la gestione dei file su disco,

```
* SP-DOS Ver 1.0 *
© 1984 SANDY PERSONAL PRODUCT.
MILANO ITALY.
```

Fig.2

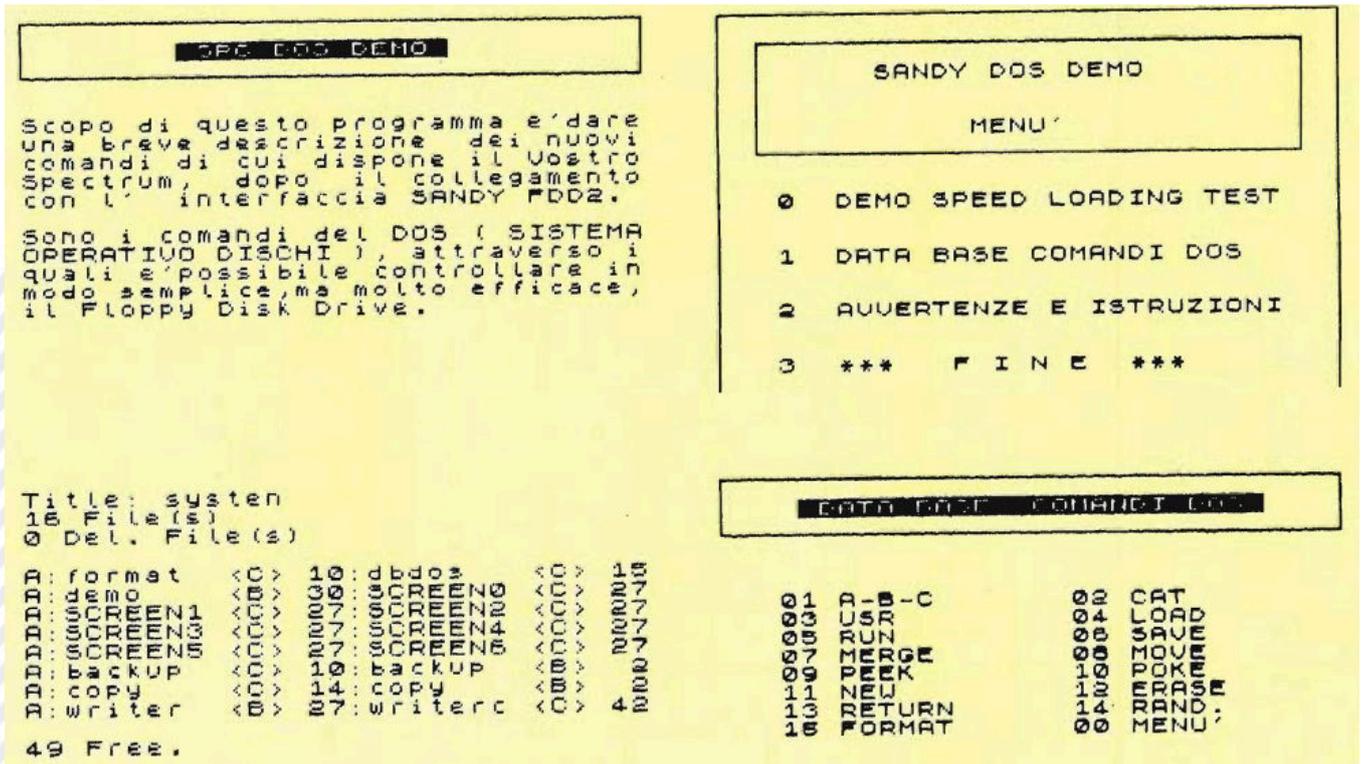


Fig.3





### **SANDY DISCO VERS.3**

Floppy disk: 3.5 pollici

Capacità: 1MB, formattato 800KB

Tracce: 80 su 2 lati

Velocità trasf.: 250 Kbit/s

Velocità LOAD/SAVE: 15 KByte/s



Fig.4

### **OPUS DISCOVERY**

1 Floppy disk drive da 3.5 pollici  
(250 KB doppia densità)

1 connettore per drive #2

1 interfaccia parallela

1 interfaccia RS-232

1 interfaccia joystick

(compatibile Kempston e Atari)

1 interfaccia video-monitor

1 connettore esterno

1 alimentatore incorporato



Fig.5

l'interfaccia Sandy metteva a disposizione (tramite una ROM "fantasma") diversi comandi BASIC. Alcuni erano derivati dal BASIC Sinclair (SAVE, LOAD, MERGE, CAT, FORMAT, ecc.) altri erano aggiunti (INIT, PUT, GET, ecc.). Molto utile era il programma dimostrativo fornito assieme alla interfaccia (fig. 3).

Successivamente fu messa in commercio una versione più evoluta con floppy disk da 3.5 pollici [Spe85] (fig. 4) che possedeva numerose interfacce (RS232 TTL, Centronics, Kempston), una uscita uscita monitor Plug RCA e un connettore da 56 pin. Il floppy disk era separato ed autolimentato e poteva essere collegato a schede previste

per altri modelli come il Sinclair QL, MSX, Amstrad, ecc.

Per chi desidera approfondire l'argomento, si trova una ottima recensione scritta (solo in italiano) da Giancarlo Errichelli per la rivista italiana "Sperimentare con l'Elettronica e il Computer" [Err85].

### **OPUS DISCOVERY**

Nel 1985, Opus Supplies Ltd (una azienda britannica di Redhill nel Surrey) mise in commercio, al prezzo di 199.95 Sterline, una interfaccia per floppy disk da 3.5 pollici da 250Kb di capacità (fig. 5). Con Discovery erano fornite utility per operazioni di "format" e "verify" e gestione "RAM DISC". Inoltre, era compatibile con i MicroDrive della ZX Interface 1 (il

DOS Discovery manteneva i comandi previsti della ZX Interface 1). Le prestazioni erano inferiori alla ZX Interface 1 (15 Kbaud contro 19.2) ma l'enfasi era posta sulla maggiore robustezza dei floppy disk rispetto ai MicroDrive.

### **DISCIPLE**

DISCIPLE era una interfaccia per il Sinclair ZX Spectrum (fig. 6,7) progettata dalla Miles Gordon Technology. Dal 1986 fu commercializzata dalla azienda britannica Rockfort Products (con sede a Londra). Somigliante esteriormente alla ZX Interface 1, era proposta come una super interfaccia con molte funzioni. Disponeva di una porta parallela Centronics, un bus Shugart per i





floppy disk, due collegamenti joystick (compatibili Kempston e Atari), un connettore audio da 3.5 mm e una connessione per ZX Interface 1. Inoltre aveva un bottone "magico" per ottenere gli snapshot. Uno snapshot è una "fotografia" dello stato della macchina (memoria e registri) che si poteva salvare su file (molto utile nei giochi per ripartire da una situazione salvata in precedenza).

I comandi per gestire i file (estensioni del BASIC Sinclair) costituivano il GDOS. Purtroppo, essendo molto sofisticata era anche costosa (73 Sterline + VAT, senza i floppy drive). Uno speciale pulsante poteva "inibire" la scheda (se necessario) per permettere il corretto funzionamento di altre periferiche.

**CHEETAH SWEET TALKER**

Sweet Talker (fig. 8) era un sintetizzatore vocale da connettere al Sinclair ZX Spectrum, prodotto da Cheetah Marketing LTD (una azienda britannica con sede a Londra). Il suo funzionamento era basato sui 64 allofoni della lingua inglese (fig. 9). Gli allofoni sono le unità di suono fondamentali nelle quali può essere scomposta qualsiasi parola. Combinandoli opportunamente, si poteva disporre di un vocabolario potenzialmente illimitato.

Un tale sistema aveva però due svantaggi. Il primo consisteva in una qualità sonora non eccelsa, il secondo era causato dagli allofoni della lingua inglese (la pronuncia di parole appartenenti ad altre lingue aveva un simpatico accento britannico). Infine mancava di un vero software di gestione e non era compatibile con i MicroDrive.

**STAMPANTE SEIKOSHA GP-50S**

Seikosha era un ramo (assorbito nel 1996) della azienda giapponese Seiko. Negli anni '80 del XX secolo produceva anche stampanti economiche.

Per i modelli Sinclair ZX81 e ZX Spectrum mise sul mercato la



Fig.6



Fig.7



Fig.8

stampante a impatto di matrice di punti GP 50 S (fig. 10). Chi scrive, fu felice di riceverne una in regalo dai propri genitori come premio scolastico, sfruttandola al massimo delle sue possibilità.

L'alimentazione era separata e il cavo di connessione al computer

un po' corto. Tuttavia il connettore lasciava spazio per aggiungere altre periferiche. Era suggerito di connettere la stampante e accenderla prima di accendere il computer.

La cartuccia della stampante conteneva un nastro da 40 cm





rinfrescato da un tampone inchiostro e si poteva utilizzare un rotolo di carta comune da calcolatrice larga almeno 3.5 pollici (circa 9 cm).

La stampante consentiva di stampare listati su 32 colonne e ottenere COPY di grafici (anche se, per esempio, i cerchi apparivano in forma ovale). Essendo anche lenta e rumorosa, non era certamente una stampante per professionisti ma era l'ideale per uno studente possessore di uno ZX Spectrum.

Tavola degli allofoni

Decimal Address	Allophones	Sample Word	Duration	Decimal Address	Allophones	Sample Word	Duration
0	PA1	PAUSE	10MS	32	/AW/	Out OU	370MS
1	PA2	PAUSE	30MS	33	/DD2/	Do D	160MS
2	PA3	PAUSE	50MS	34	/GG3/	Wig IG	140MS
3	PA4	PAUSE	100MS	35	/VV/	Vest V	190MS
4	PA5	PAUSE	200MS	36	/EG1/	Guest GU	80MS
5	/OY/	Boy OY	420MS	37	/SH/	Ship S	160MS
6	/AY/	Sky Y	250MS	38	/ZH/	Azure Z	190MS
7	/EH/	End E	70MS	39	/RR2/	Brain R	120MS
8	/KK3/	Comb C	120MS	40	/FF/	Food F	150MS
9	/PP/	Pow P	210MS	41	/KK2/	Sky K	190MS
10	/JH/	Dodge G	140MS	42	/KK1/	Can't C	160MS
11	/NN1/	Thin N	140MS	43	/ZZ/	Zoo Z	210MS
12	/IH/	Sit I	70MS	44	/NG/	Anchor N	220MS
13	/TT2/	To T	140MS	45	/LL/	Lake L	110MS
14	/RR1/	Rural R	170MS	46	/WW/	Wool W	180MS
15	/AX/	Succeed U	70MS	47	/XR/	Repair R	360MS
16	/MM/	Milk M	180MS	48	/WH/	Whig W	200MS
17	/TT1/	Part T	100MS	49	/YY1/	Yes Y	130MS
18	/DH1/	They TH	290MS	50	/CH/	Church C	190MS
19	/IY/	See E	250MS	51	/ER1/	Fir IR	160MS
20	/EY/	Beige EI	280MS	52	/ER2/	Fir ERR	300MS
21	/DD1/	Could ID	70MS	53	/CW/	Beau AU	240MS
22	/UW1/	To O	100MS	54	/DH2/	They TH	240MS
23	/AO/	Aught AU	100MS	55	/SS/	Vest S	90MS
24	/AA/	Hot O	100MS	56	/NN2/	No N	190MS
25	/YY2/	Yes YE	180MS	57	/HH2/	Hoe H	180MS
26	/AE/	Hat A	120MS	58	/OR/	Store OR	330MS
27	/HH1/	He H	130MS	59	/AR/	Alarm A	290MS
28	/BB/	Business BU	80MS	60	/YR/	Clear R	350MS
29	/TH/	Thin TH	180MS	61	/EG2/	Got G	40MS
30	/UH/	Book OO	100MS	62	/EL/	Saddle L	190MS
31	/UW2/	Food OO	260MS	63	/BB2/	Business B	50MS

Fig.9

### SEIKOSHA GP 50 S

Modo di stampa: impatto a matrice di punti (8x8)  
 Testina: a martello singolo  
 (garantiti 30 mln di caratteri)  
 Formato: 32 colonne  
 Interlinea: 1/9 pollici  
 Velocità: 35 caratteri al secondo (dichiarati)  
 Spazio orizz. tra i punti: 1/84 pollici  
 Spazio vert. tra i punti: 1/72 pollici  
 Avanzamento carta: 6.6 linee per sec.  
 Trascinamento: a frizione  
 Alimentazione: trasformatore 220V AC-18V DC  
 Consumo: 11W in stand-by, 17W durante la stampa  
 Dimensioni: 85mm x 250mm x 215mm  
 Peso: 1.5 Kg



Fig.10

### Bibliografia

- [Dis21] AA.VV., "DISCiPLE", retrieved (2021-12-10) from <https://en.wikipedia.org/wiki/DISCiPLE>
- [OD21] Spectrum Computing retrieved (2021-12-10) from <https://spectrumcomputing.co.uk/index.php?cat=96&id=1000297>
- [Err85] G. Errichelli, "Floppy Disk Sandy", *Sperimentare con l'elettronica e il Computer*, Jul-Aug 1985, pp. 40-43, [http://www.introni.it/pdf/Sperimentare%201985\\_07\\_08.pdf](http://www.introni.it/pdf/Sperimentare%201985_07_08.pdf)
- [Ber84] M. Bergami, "CHEETAH Sweet Talker", *MC Microcomputer* n.29, Apr. 1985, pp. 68-70, [http://www.dizionariovideogiochi.it/doku.php?id=mc\\_-\\_029](http://www.dizionariovideogiochi.it/doku.php?id=mc_-_029)
- [SC84] AA.VV., "Seikosha GP50S", *Sinclair Computer* n.7, Oct 1984.
- [SCo21] AA.VV., "Disciple Interface", retrieved (2021-12-10) from [https://spectrumcomputing.co.uk/entry/1000117/Hardware/DISCiPLE\\_Interface](https://spectrumcomputing.co.uk/entry/1000117/Hardware/DISCiPLE_Interface)
- [Spe85] *Sperimentare* Dec. 1985, pp 8-9, retrieved (2021-12-10) from [http://www.introni.it/pdf/Sperimentare%201985\\_12.pdf](http://www.introni.it/pdf/Sperimentare%201985_12.pdf)
- [Wa87] J. Wase, "Disciple", *ZX Computing Monthly*, Mar. 1987, pp.30-31.
- [WS21] World of Spectrum retrieved (2021-12-10) from <https://worldofspectrum.org/hardware/>





# ColecoVision

di Leonardo Miliani

Seguendo il filo delle console che hanno segnato più di altre il settore dei videogiochi casalinghi, in questo articolo andremo ad analizzare un oggetto appartenente alla seconda generazione dei dispositivi da gioco domestici, nato, come la Fairchild Channel F analizzata la volta precedente, sulla East Coast americana. Questa console detiene un paio di primati: è la prima nel suo genere che abbia offerto giochi con una grafica pari a quella degli arcade dell'epoca, ed è la prima che abbia offerto delle espansioni hardware che modificano anche radicalmente le sue caratteristiche di base. Stiamo parlando del ColecoVision (fig. 1).

## Un po' di storia

Per capire la genesi dei dispositivi analizzati nei precedenti articoli, quasi sempre siamo partiti diversi anni prima rispetto al loro arrivo sul mercato ma questa volta esageriamo, e non poco: questa volta torniamo indietro nel 1932! Siamo a West Hartford, una cittadina di qualche decina di migliaia di persone nel centro del Connecticut. Maurice Greenberg fonda la Connecticut Leather Company, una società che vende pelli e strumenti per la loro lavorazione ai ciabattini che riparano le scarpe. Nel 1938 inizia a produrre e vendere anche stivali in pelle. Con l'entrata degli Stati Uniti d'America nella Seconda Guerra Mondiale gli affari decollano per l'aumento delle richieste di questi articoli. Nel 1945 l'offerta di vendita è ricca: macchinari per la lavorazione delle scarpe, cappelli e accessori per la loro pulizia, e postazioni per i lustrascarpe. Nel giro di qualche anno, grazie alle idee del figlio Leonard, la società di Greenberg inizia ad offrire anche kit per la realizzazione in proprio di articoli in pelle, spostandosi dalla vendita all'ingrosso a quella al dettaglio.

Ma cosa c'entra tutto questo con le console da gioco? È

proprio l'aver abbracciato la vendita al consumatore finale che nel giro di pochi anni farà cambiare radicalmente settore di mercato alla società. Nel 1954, infatti, durante il New York Toy Fair, un evento dedicato al mondo dei giocattoli, il kit per la realizzazione di mocassini in pelle vince il premio per il miglior giocattolo pedagogico per bambini! Grazie a questo premio, Leonard Greenberg sposta gli interessi della società verso il settore dei giocattoli, arrivando poi a venderne nel 1961 il ramo aziendale degli articoli in pelle e cambiandone il nome in Coleco Industries Inc. Nel 1963 Coleco entra nel mondo delle piscine fuori terra divenendone il più grosso produttore al mondo grazie all'acquisizione di un'importante azienda del settore. Nel 1966 entra nella società anche il fratello di Leonard, Arnold, che, suggerendo ulteriori acquisizioni, guida l'azienda verso l'affermazione nel settore dei giocattoli. Agli inizi degli anni '70 la Coleco entra anche nel mercato delle motoslitte da neve ma gli affari non vanno molto bene, ed i profitti calano. Arnold Greenberg, divenuto amministratore delegato, nel 1976 decide di entrare nel mondo delle console da gioco attratto dal successo del Pong di Atari e di tutti i suoi cloni, che hanno letteralmente invaso le case degli americani.

## Gli antenati del ColecoVision

Nel 1976 Coleco rilascia il Telstar (fig. 2), una semplice console da collegare al televisore con 2 paddle integrati basato sul "Pong-on-a-chip" di General Instruments, un unico integrato che offre 3 giochi in stile Pong/Tennis, che diviene un immediato successo: ne vengono vendute più di 1 milione di unità. Però, l'interesse per questi dispositivi cala rapidamente, messi in ombra dalle nuove console programmabili in uscita alla fine degli anni '70, prima fra tutte l'Atari VCS/2600. Questo porta Coleco quasi alla bancarotta ma la società riesce a riprendersi producendo dei piccoli videogiochi portatili come l'Electronic Quarterback, un gioco dedicato al football. Grazie alla qualità di questi videogiochi, Coleco ruba la prima piazza alla Mattel, che fino a quel momento è stata la leader indiscussa di quella nicchia di mercato. Forte di questo nuovo successo, nel 1979 Coleco inizia lo sviluppo di una console domestica vera e propria. Eric Bromley, un



Fig. 1: Il ColecoVision (foto: Evan-Amos – fonte: Wikimedia Commons)





**Fig. 2 - Il Coleco Telstar (foto: Jzh2074 – fonte: Wikimedia Commons)**

ingegnere che ha lavorato in passato per altri produttori di arcade tra cui Midway, viene assunto per sviluppare un sistema basato su cartucce come mezzo per la distribuzione dei giochi. La qualità del sistema è molto buona ma le richieste hardware sono notevoli e Greenberg ferma lo sviluppo perché il notevole quantitativo di RAM richiesto rende i costi di produzione troppo elevati (non dimentichiamoci che alla fine degli anni '70 del secolo scorso le RAM erano molto costose tant'è che i primi home computer avevano piccole dotazioni di memoria, nell'ordine mediamente di 1, 2, o 4 KB). Le cose cambiano nel 1981: i prezzi delle RAM iniziano a scendere e Bromley si ripresenta da Greenberg facendo vedere che adesso è possibile realizzare la sua console e metterla in commercio ad un prezzo competitivo. Greenberg autorizza la ripresa dello sviluppo del dispositivo ma capisce anche che la console avrà bisogno di un forte traino per inserirsi nella mischia, all'epoca dominata dall'Atari 2600 e dall'Intellivision di Mattel. Coleco, che nel frattempo aveva continuato ad investire sul mercato dei videogiochi, agli inizi del 1982 immette sul mercato i Mini-Arcade, delle piccole riproduzioni dei giochi arcade più in voga in quel periodo ma basati su componenti elettronici a stato solido (tipo le console di prima generazione) che riproducono (in maniera molto limitata) la modalità di gioco dell'arcade replicato, sia esso Pac-Man, Frogger o Galaxian. Questi oggetti riscuotono un enorme successo, con vendite nell'ordine dei milioni di unità. Uno dei fattori di questo successo è il fatto di poter usare sotto licenza il nome dei titoli più in voga nelle sale giochi. Greenberg e Bromley decidono perciò di allegare alla console un gioco di successo in esclusiva, non solo per mostrarne le capacità grafiche ma anche per costringere chi vuole giocare a quel titolo a casa propria ad acquistare quella console.

### La trattativa con Nintendo

In quel periodo i maggiori giochi arcade arrivano dal Giappone e Greenberg, agli inizi del 1982, spedisce Bromley a Kyoto a parlare direttamente con il presidente di Nintendo Hiroshi Yamauchi per trattare l'esclusiva di un titolo di successo. Le trattative vanno avanti con alterne vicende: Yamauchi propone giochi di scarso valore, che sicuramente non attirerebbero molta attenzione. Ma la fortuna sta per metterci una pezza: durante una pausa pranzo, l'allontanarsi per un semplice bisogno fisiologico porta Bromley alla scoperta dell'oro! Mentre infatti cerca il bagno, passa davanti ad una stanza al cui interno c'è un cabinet con sui fianchi disegnati una grossa scimmia ed un gioco che lo attira subito: «Donkey Kong» (fig. 3). Quello è il gioco di cui Coleco ha bisogno! Le trattative con Yamauchi vengono subito spostate verso Donkey Kong e Bromley si accorda per la sua esclusiva: l'accordo prevede il pagamento immediato di 200.000 dollari come costo della licenza e di 2 dollari di royalty per ogni cartuccia venduta. Bromley torna in America con un accordo firmato su dei tovagliolini di carta... perché all'epoca in Giappone i contratti erano per lo più accordi verbali.

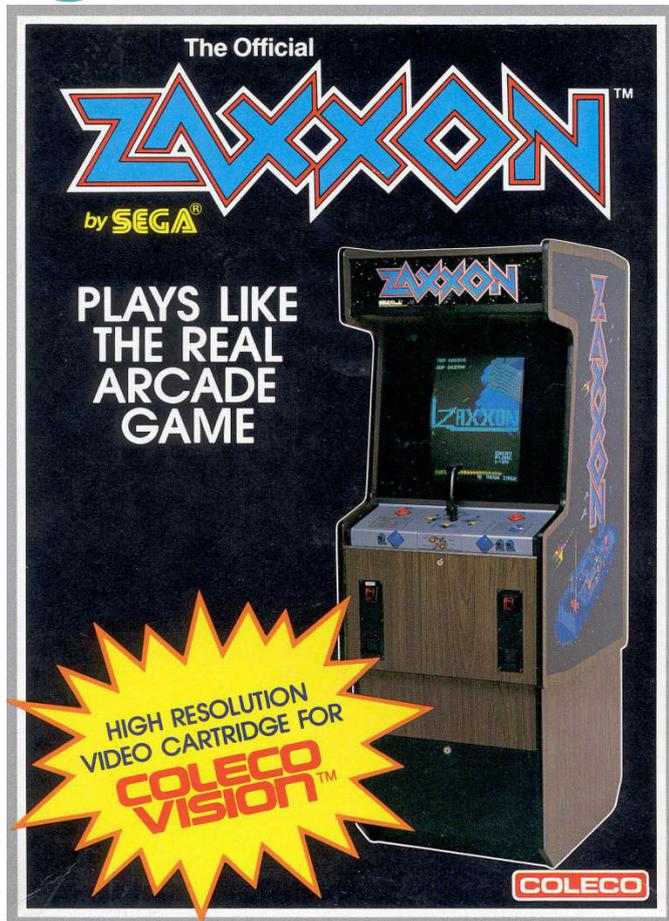
Ma le cose stanno per prendere una brutta piega. Durante il Consumer Electronic Show di giugno, Bromley scopre che allo stand Atari viene mostrato Donkey Kong! Bromley viene a sapere dalla figlia di Yamauchi, che fa da interprete durante le trattative col padre, che il presidente di Nintendo, volendo entrare nel mercato dei videogiochi americano, si è accordato con Atari, allora uno dei più grossi nomi del settore, per commercializzare il suo nuovo gioco. Bromley, con l'aiuto della figlia, riesce a trattare un nuovo accordo (questa volta firmato) con il quale a Coleco viene concesso Donkey Kong in esclusiva per le console, mentre restano ad Atari i diritti per i computer.

### La commercializzazione ed i moduli di espansione



**Fig. 3 - Donkey Kong**





**Fig. 4 - Copertina della cartuccia del gioco "Zaxxon". Si noti l'enfasi posta sul fatto che il gioco è fedele all'originale (foto: Servo - fonte: Mobygames.net)**

La nuova console viene messa in commercio durante il mese di agosto del 1982 come ColecoVision al prezzo di lancio di 174,99 dollari: ogni confezione contiene al suo interno il gioco Donkey Kong. Altri giochi seguono poi il lancio della macchina. Coleco pubblicizza su riviste e giornali le qualità tecniche del ColecoVision, di molto superiori a quelle dei suoi diretti rivali Atari 2600 e Mattel Intellivision: per la prima volta le conversioni sono molto fedeli agli arcade dei bar e delle sale giochi (fig. 4). Donkey Kong è praticamente uguale all'originale, Lady Bug, Turbo e Smurf (i "Puffi") sorprendono, Zaxxon, con la sua prospettiva isometrica, colpisce tutti.

Tutto ciò è possibile perché il ColecoVision è la prima console ad essere basata su una CPU normalmente impiegata sui computer: usa infatti l'affermato Zilog Z80 che muove la stragrande maggioranza delle macchine ad 8 bit con sistema operativo CP/M, che offre una potenza di calcolo di molto superiore ai "cervelli elettronici" che muovono le altre console. Ad esso si affianca il chip video TMS9928A, variante del più noto TMS9918A che è usato nel Texas Instruments TI-99/4A e che poi diverrà il cuore dei sistemi MSX e di altri computer e console, che offre un'immagine di 256x192 pixel a 16 colori con la bellezza

di 32 sprite con risoluzioni da 8x8 a 32x32 bit. Il suono è generato dal Texas Instruments SN76489A, largamente usato in computer, console e arcade: offre 3 canali audio ed 1 canale di rumore bianco. Lato memoria il sistema è riccamente dotato: 8 KB di ROM contenente il BIOS della macchina, 1 KB di RAM per i dati dei programmi e 16 KB di VRAM esclusivamente dedicati al processore video per memorizzare i dati dell'immagine e quelli degli sprite. Il design della console è molto curato: è un unico assemblato in plastica nera basso e filante, con un alloggiamento superiore per riporre i gamepad quando non in uso. Questi ultimi sono dotati di un joystick e di un tastierino numerico usato per comandi aggiuntivi dei vari giochi, su cui si applicano delle didascalie fornite dai produttori delle cartucce per ricordare più facilmente il significato dei comandi. Una nota che stride in tutto questo è l'inspiegabile ritardo di 12 secondi, dall'accensione della console, prima dell'avvio effettivo del gioco, durante il quale la console mostra solo il logo "ColecoVision" sullo schermo: non è un difetto, è una precisa (e scellerata) scelta commerciale di Coleco (quando verrà introdotta sul mercato europeo, tale ritardo sarà ridotto a 3 secondi). Alcuni produttori di giochi trovano comunque il modo di bypassare questo ritardo del BIOS e far partire subito i propri giochi.

Entro la fine dell'anno Coleco vende 500.000 console, che diventano 1.000.000 agli inizi del 1983. In quell'anno arrivano altri titoli come Buck Rogers, Centipede, Defender, Donkey Kong Jr., Frogger, Galaxian, Pitfall!, Popeye, Time Pilot ed altri, seguiti poi nel 1984 da altri titoli di successo come BurgerTime, Choplifter, Gyruss, Q\*berts, River Raid, Spy Hunter. Le vendite aumentano con l'inizio della distribuzione al di fuori degli USA: a luglio del 1983 la console fa il suo debutto in Europa, distribuita da CBS Electronics come «CBS ColecoVision».

Ciò che rende appetibile il ColecoVision sono i suoi «Expansion Modules», o moduli di espansione, accessori da collegare generalmente alla porta di espansione della console che ne modificano anche sostanzialmente le caratteristiche. L'«Expansion Module #1» trasforma il ColecoVision in un Atari 2600 grazie al fatto che al suo interno è presente un clone perfetto di questa console. Questo modulo permette al ColecoVision di poter accedere alla vastissima libreria di giochi dell'Atari 2600. Atari cita anche in tribunale Coleco per questo accessorio ma perde la causa perché all'epoca le leggi americane in merito ai brevetti sui videogiochi erano un po' lacunose e siccome il modulo era fatto ricorrendo a componenti comunemente reperibili in commercio, il verdetto è stato che a norma





**Fig. 5 - L'Expansion Module #2**  
(foto: Evan Amos - fonte: Wikimedia Commons)

di legge nessun componente proprietario era stato usato senza licenza (grazie a questa sentenza, Coleco mette in commercio l'adattatore anche come console a sé stante con il nome di «Coleco Gemini»). L'«Expansion Module #2» (fig. 5) è essenzialmente un volante con pedaliera inizialmente offerto con la cartuccia «Turbo». Il pedale del gas è un semplice interruttore on/off, non offrendo quindi un controllo analogico del gas. È stato usato anche da altri giochi di guida ma non ha avuto molta fortuna commerciale. L'«Expansion Module #3» è l'espansione più interessante. Dato che nella prima metà degli anni '80 si stanno affermando sempre più gli home computer, Coleco decide di cavalcare l'onda e, forte del fatto che la sua console integra una CPU ad 8 bit normalmente usata sui computer, crea questa espansione che trasforma il ColecoVision nell'«Adam Computer» (fig. 7). Si compone di un corpo macchina a cui va agganciata posteriormente la console e che contiene, oltre alla scheda madre con 64 KB di RAM, anche un lettore di nastri magnetici, di una



**Fig. 6 - Il Super Action Controller**  
(foto: Miguel Duran - fonte: Museo 8 bits)

stampante, di una tastiera e della cartuccia del gioco «Buck Rogers: Planet of Zoom» (non può usare Donkey Kong perché i diritti per questo gioco sui computer, come scritto in precedenza, li detiene Atari). A parte può essere acquistata un'unità a dischi. Nonostante le potenzialità, i vari difetti che lo affliggono e l'elevato costo ne limitano il successo.

Oltre a questi moduli vengono offerti anche dei controller avanzati. Il «Roller Controller» è un trackball offerto con il gioco «Slither» che, assurdamente, usa un connettore non compatibile con il computer Adam per cui Coleco si vede costretta a spedire l'adattatore a chi ne fa richiesta. Il «Super Action Controller» (fig. 6) è invece un set di 2 joystick speciali offerti inizialmente con la cartuccia «Super Action Baseball». Ogni dispositivo ha un joystick con top a sfera sulla parte superiore, dove è presente anche una tastiera con 12 pulsanti e uno speeder roller, un potenziometro a rotella per controllare la velocità della palla; nell'impugnatura sono presenti 4 pulsanti aggiuntivi. Questo speciale controller è stato usato anche da altri giochi. Da segnalare che in origine Coleco aveva iniziato la progettazione di un modulo denominato «Super Game Module» (SGM), al cui sviluppo aveva partecipato in un secondo momento anche Ralph Baer, l'ideatore della prima console della storia, il Magnavox Odyssey. Il modulo avrebbe dovuto permettere di poter caricare dei giochi da un lettore a nastri o dischi. L'idea è stata accantonata quando fu deciso di sviluppare il modulo Adam.

### La crisi del 1983 e l'uscita di scena

La console, come detto, vende veramente bene. La qualità dei suoi giochi, forti delle capacità hardware della macchina, è superiore a quella delle rivali e le sue conversioni sono realmente paragonabili come aspetto e giocabilità agli arcade da cui derivano. Ciò che rovina un po' l'immagine del ColecoVision sono i vaporware di cui soffre Coleco: troppe volte vengono presentati giochi o prodotti che poi subiscono notevoli ritardi o addirittura vengono cancellati e mai messi in commercio. Uno di questi vaporware è il già menzionato SGM: annunciato per il mese di agosto del 1983, viene prima ritardato, poi rivisto, e successivamente cancellato, sostituito dal modulo Adam. Anche questo subisce la stessa sorte nefasta: presentato nel mese di giugno del 1983, doveva arrivare a breve e vendere, secondo i piani, 500.000 pezzi entro Natale. L'uscita viene però rimandata di mese in mese, e l'Adam arriva in commercio solo a metà ottobre, con un prezzo lievitato dai 525 dollari iniziali ai 725 finali. Se a questo





**Fig. 7 - Il Coleco Adam (foto: Doontass - fonte: Wikimedia Commons)**

importo si somma il costo di acquisto della console si arriva ad una cifra superiore a quella necessaria all'acquisto del ColecoVision e di un Commodore 64, il computer che in quel periodo si sta imponendo in maniera massiccia per le sue doti grafiche e sonore. L'Adam è poi afflitto da gravi difetti di fabbricazione: molte unità si guastano dopo un breve utilizzo, altre addirittura sono difettose già di fabbrica, con il risultato che i resi per le riparazioni sono elevati. A causa di ciò entro il 1983 ne vengono vendute solo 100.000 unità, con una perdita stimata di 35 milioni di dollari.

Durante il 1983 avviene poi un altro fatto che mina le finanze di Coleco: si compie la famosa "crisi dei videogiochi", che colpisce duramente il settore, portando le aziende più piccole al fallimento e quelle più grandi ad un massiccio ridimensionamento. La stessa Atari, che ha dominato fino ad allora il settore dei videogiochi, quell'anno accusa perdite per mezzo miliardo di dollari, rischiando la chiusura. Anche Coleco subisce la crisi: nonostante il ColecoVision continui a vendere discretamente (arriva a 2 milioni di unità nel 1984), le perdite causate dai ritardi e dai difetti dell'Adam e lo scarso commercio delle cartucce per via della crisi portano la casa madre a rientrare sul mercato dei giocattoli per cercare di fare cassa. A fine 1983 presenta una serie di pupazzi in stoffa denominati «Cabbage Patch Kids», che vendono bene e riescono a tenere a galla la società. Il 1984 termina con altri 80 milioni di dollari di perdite, dovuti prevalentemente ai problemi legati all'Adam: agli inizi del 1985 la società decide alla fine di toglierlo dal mercato mentre entro la fine dell'anno cessa le vendite

anche del ColecoVision. I numeri di vendita totali della console non sono noti con esattezza ma si stima che complessivamente siano stati commercializzati 6 milioni di ColecoVision: cifre non eclatanti ma sicuramente interessanti, visto il periodo in cui la console è stata messa in vendita, avendo dovuto cavalcare l'onda della crisi del settore che ha colpito l'America.

In tempi più recenti la console ha vissuto una seconda giovinezza grazie a diversi sviluppatori indipendenti che hanno rilasciato nuovi giochi e, grazie alle somiglianze con i sistemi MSX e Sega SG-1000, fatto il porting di numerosi giochi di queste due piattaforme. Non solo, nel 2014 è stata messa in vendita la «ColecoVision FlashBack» (fig. 8), una versione moderna della console con 60 titoli pre-installati per far assaporare alle nuove generazioni il gusto dei giochi degli anni '80: un modo per onorare un sistema che avrebbe potuto avere tanto



**Fig. 8 - ColecoVision FlashBack (fonte: AtGames)**





# RetroFighter engine per Commodore 64

di Gianluca Alberico

Mi chiamo Gianluca Alberico, ho 42 anni, di Roma, diploma di perito industriale. Sono un programmatore congenito: programmo dall'età di 10 anni, ho cominciato con un C64, per poi proseguire con Amiga e infine PC. Ho da sempre la passione per la programmazione di videogame e della computer graphics in generale. Dopo il diploma creai un mio sito Internet, con alcuni giochi sviluppati da me per MS-DOS. Con il passare degli anni e a causa di alcune scelte di vita, sono poi diventato un programmatore .NET/Xamarin, ma nel tempo libero mi metto sempre a fare qualcosa relativamente alla programmazione di giochi.

## Le ragioni che mi hanno spinto a creare l'engine

Verso la fine del 2021 mi era venuta voglia di giocare a qualche picchiaduro per C64, magari pubblicato di recente, ma mi sono accorto che ancora non ce n'erano che presentassero una giocabilità pari o simile a quelli che generalmente uscivano per le console dell'epoca (SNES, Genesis, ecc.). Avevo notato anche la demo su YouTube di SF2CE ad opera di Pacoblog64 [1], ma non mi aveva soddisfatto. Quindi agli inizi di ottobre 2021, dopo alcune ricerche, mi ero posto una sfida tecnica: è possibile creare, su C64, un engine per picchiaduro che faccia qualcosa di più di quanto visto finora? E quindi ho iniziato a sviluppare un motore, inizialmente per PC, in linguaggio C, che riproducesse per prima cosa la giocabilità di Street Fighter 2 ma con sprite adatti alla conversione su C64. Poi ho preso a convertire sia la grafica sia il codice per C64 e, risolvendo un problema alla volta, sono riuscito a produrre quello che già è presente nel mio canale [2].

## In cosa consiste l'engine?

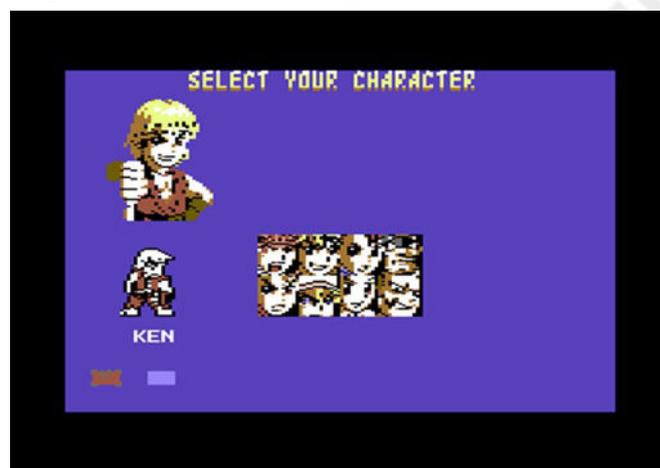
L'engine è in grado di implementare una sessione di combattimento all'interno di un picchiaduro tra due giocatori. Il cuore dell'engine, cioè quello che rappresenta la sessione di gioco vera e propria, è sviluppato in linguaggio C per permettere una portabilità futura del motore stesso su altre piattaforme (ad es. Amstrad CPC). La struttura dell'engine ricorda un po' quella del MUGEN [3], ma ovviamente le sue potenzialità sono molto ridotte rispetto a quest'ultimo in quanto deve essere eseguito su sistemi

ad 8 bit (per ora) con risorse hardware limitate.

## Cosa può fare

L'engine ha le seguenti caratteristiche:

- Ogni combattente può attingere fino a 63 frame con dimensione variabile. Non ci sono vincoli di disegno all'interno dei frame (essi possono essere disegnati così come si disegnerebbero i frame dei giocatori di un gioco analogo per PC), l'importante è rispettare la limitazione del C64 di avere 3 colori più 1 di trasparenza. L'engine implementa una tecnica di sprite multiplexing per permettere l'estensione degli sprite in verticale; tuttavia, per limitare il consumo di memoria, l'engine permette al momento una dimensione massima dei frame di 48x63 pixel.
- Ogni combattente è definito da una collezione di sequenze. Ogni sequenza è uno script che può eseguire diversi comandi (spostamento, lancio di proiettili, effetti vari, ecc.). Esistono al momento quasi una quarantina di sequenze definibili per ogni combattente, ma tra un po' di tempo sarò forse in grado di definirne di più (circa sessanta) e soprattutto più lunghe.
- Ogni combattente può eseguire, al momento, 6 mosse base a terra, 2 mosse volanti, 3 mosse speciali e una super-mossa. Tra un po' di tempo, sarò forse in grado di aumentare il numero di mosse base, speciali e super-mosse da eseguire.
- Ogni combattente può eseguire un solo tipo di presa. Come prima, tra un po' di tempo, sarò forse in grado di aumentare il numero di prese da eseguire.
- Ogni combattente può eseguire i colpi che si trovano



Selezione del personaggio con relativo colore





### Colpi infuocati e mosse speciali

più comunemente in tutti i picchiaduro: colpi deboli e forti, colpi che stordiscono, colpi che immobilizzano, colpi infuocati.

- Gli sfondi hanno una dimensione fissa di 320x112 pixel, e devono essere character-based (non possono essere disegnati liberamente pixel per pixel ma devono essere composti da una serie di blocchi provenienti da una palette personalizzata). Gli sfondi possono essere animati: animazioni semplici ovviamente, che consistono nel cambiare semplicemente il contenuto di una parte dello schermo con altro contenuto in maniera periodica.

- Attualmente non sono implementati ma nella prossima demo ci sarà la possibilità di creare dei bonus round molto semplici, in cui il giocatore dovrà distruggere uno o più oggetti animati o non.

- Possono essere definiti, per ogni combattente, due "filmati" che verranno mostrati nello story mode: introduttivo e conclusivo. Quello introduttivo verrà mostrato dopo la selezione del personaggio, quello conclusivo verrà mostrato dopo che verrà battuto l'ultimo avversario. Ogni filmato è anch'esso uno script e può attingere a diverse bitmap (quindi può contenere ciò che si vuole, senza essere vincolati ai blocchi ma rispettando le restrizioni dei colori

del C64). Lo stesso linguaggio di scripting può essere utilizzato per creare l'animazione iniziale del gioco quando quest'ultimo viene avviato.

- Per ogni combattente è possibile definire la lista dei suoi avversari.

- Per ogni combattente è ovviamente possibile definire la lista delle frasi caratteristiche di vittoria e provocazione.

- E' possibile definire i colori che il giocatore può scegliere per ogni combattente.

### Cosa non può fare

Al momento:

- L'engine non fornisce un motore interno per suonare una musica di sottofondo ma può solo riprodurre semplici effetti sonori.

- L'engine non è in grado di definire mosse speciali volanti.

- L'engine non è in grado di permettere ad un giocatore di sparare più di un proiettile alla volta.

Sicuramente, in futuro:

- L'engine non sarà in grado di utilizzare sfondi bitmap-based in quanto, se così fosse, le animazioni degli sfondi sarebbero più lente e il frame rate troppo basso.

- L'engine prevede solo un campo di gioco ristretto e largo





### Diversi stage sia animati che non

massimo 320 pixel.

Con "al momento" non intendo dire per forza che in futuro la caratteristica relativa sarà implementata, ma solo che la prenderò in considerazione per una ulteriore valutazione (tranne ovviamente la musica che sarà inserita nel gioco da produrre con questo engine).

### Gioco da finire di implementare: SNK vs CAPCOM

L'intenzione è quella di creare al momento un gioco chiamato semplicemente "SNK vs CAPCOM", ispirato dall'omonimo gioco per Neo Geo Pocket Color. Il gioco

non sarà una copia conforme di quello per NGPC, ma semplicemente una versione personalizzata ed adattata per il C64. Sarà semplicemente un gioco sulla falsariga di Street Fighter 2 ma con i personaggi di SNK vs CAPCOM: un torneo a cui partecipano i vari combattenti, ognuno con la propria motivazione. Nel gioco saranno presenti 13 personaggi (12+1 segreto). Il gioco verrà rilasciato in formato cartuccia (file .CRT).

### Come si integra?

Al momento l'engine non è fruibile da altri programmatori perché deve essere ancora completato e devono essere separate meglio le varie parti costituenti. Tuttavia in futuro prenderò in seria considerazione la possibilità di farlo diventare un SDK da poter integrare su diverse piattaforme.

### Riferimenti

- [1] SF2CE PacoBlog64: <https://www.pacoblog64.com/2020/05/desarrollo-de-street-fighter-2-champion.html>
- [2] RetroGL - [https://www.youtube.com/channel/UCshZEsiKWs\\_xUjXdifb3T4A](https://www.youtube.com/channel/UCshZEsiKWs_xUjXdifb3T4A)
- [3] MUGEN Engine - <https://mugen.weboy.org/category/database/full-games/>



### Super-mosse speciali per tutti i personaggi





# Atari ST Basic - AES e VDI

di Francesco Fiorentini

Da ragazzo, prima di passare ai PC, ho posseduto il Commodore 64 e l'Amiga 500. Come molti dei nostri lettori ho cominciato a programmare in Basic con il Commodore 64 e poi successivamente in AmigaBASIC sull'Amiga, per poi passare ad altri linguaggi sul 16 bit di casa Commodore e sui PC IBM. Il linguaggio Basic mi ha sempre affascinato per la sua semplicità di utilizzo e soprattutto per il fatto che fosse, quasi sempre, il linguaggio presente di default sulle macchine ad 8 e 16 bit. Era naturale quindi, all'epoca, che mi interrogassi su quali fossero gli ambienti presenti sulle macchine che non possedevo e che fossi curioso di provarli. Purtroppo non essendo figlio di Kashoggi, le disponibilità dell'epoca non mi consentivano tanti voli pindarici, quindi ho sempre dovuto tenere la mia curiosità sotto controllo. Fino ad ora...

Adesso con gli emulatori posso sbizzarrirmi e provare ad utilizzare computer che non ho posseduto e soprattutto, scrivere due righe di codice in un Basic che all'epoca non potevo utilizzare. Tra tutti i computer antagonisti dei miei posseduti, ho sempre avuto un debole per l'Amstrad CPC e l'Atari ST. Ho soddisfatto la mia curiosità con l'Amstrad utilizzando il Locomotive Basic e scoprendo un linguaggio incredibilmente evoluto rispetto al povero Basic V2. I miei esperimenti con quella macchina e quel linguaggio non sono certo terminati, ma la curiosità adesso mi spinge a provare anche l'Atari ST ed il suo ST Basic. Da quello che ho letto, le funzionalità dell'ST Basic sono purtroppo paragonabili a quelle del misero Amiga Basic, quindi nessuna emozionante scoperta come con il Locomotive

Basic. Vi starete chiedendo perché allora perdere tempo a provare questo linguaggio. La risposta è semplice: fa parte della storia della macchina e serve per comprendere la ragione per cui altri Basic (GFA Basic, STOS...) sono stati creati per questa macchina (...e per Amiga). E poi, come dico sempre è divertente scoprire cose nuove.

## Atari ST Basic

La prima release dell'ST Basic è del 1985, coetaneo del TOS. Prodotto da MetaComco su richiesta dell'Atari per sfruttare l'ambiente GEM su cui si basa l'Atari ST (vedremo più avanti di cosa si tratta), si rivelò purtroppo pieno zeppo di bug e quindi inadeguato per le ambizioni della macchina ospite. Dato che era fornito a corredo con l'Atari ST, ottenne comunque un discreto seguito visto che, almeno inizialmente, riviste e produttori di software lo utilizzarono per distribuire applicazioni elementari.

## ST Basic - ambiente di sviluppo

L'ambiente di sviluppo dell'ST Basic è molto simile come concezione a quello dell'Amiga Basic, quindi chiunque abbia un minimo di familiarità con tale ambiente, si troverà presto a suo agio.

L'interfaccia si compone di 4 finestre:

- EDITOR - dove inserire il listato vero e proprio
- LIST - visualizza l'intero listato del programma
- COMMAND - dove inserire i comandi immediati (ed anche il listato)
- OUTPUT - dove viene mostrato il risultato del programma



Fig. 1 - Atari ST (immagine da Wikimedia Commons)

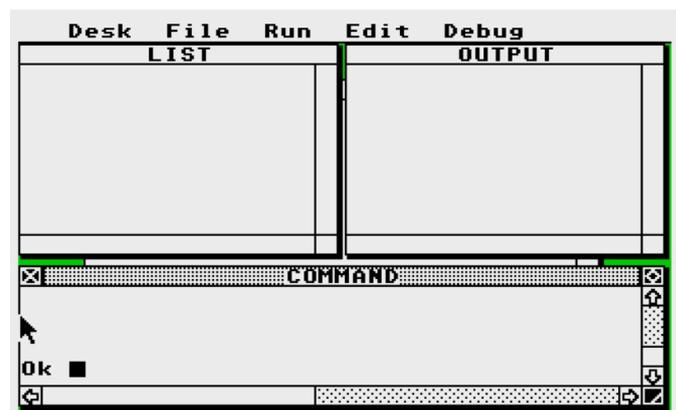


Fig. 2 - L'ST Basic all'avvio



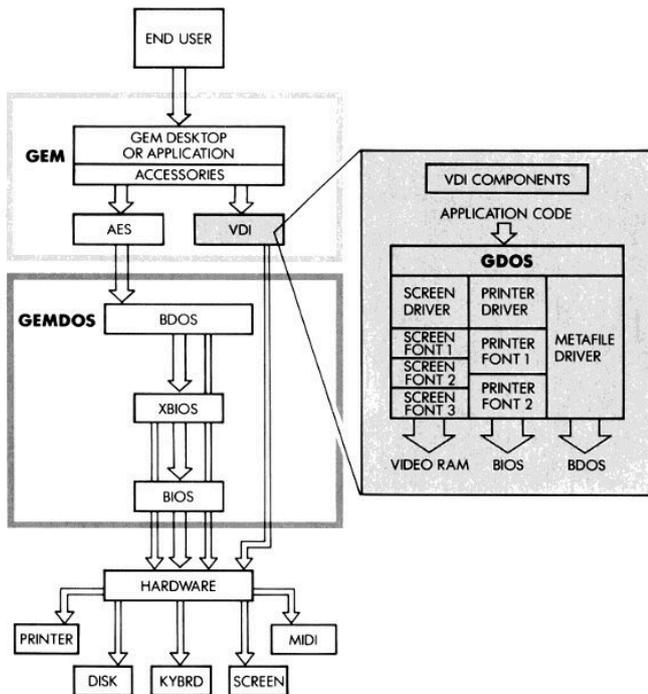


Fig. 3 - Architettura logica del TOS

All'avvio dell'ST Basic, vengono contemporaneamente mostrate 3 delle finestre sopra citate (figura 2). Date le scarse dimensioni dello schermo, non é proprio il massimo, ma con un po' di abilità impareremo a destreggiarci.

Se devo fare un paragone immediato con l'Amiga Basic, ho trovato l'ambiente ST Basic leggermente piú performante e quindi meno frustrante nel suo utilizzo.

Per ovvie ragioni di spazio e per rendere l'articolo un po' piú interessante, invece di parlare dei soliti comandi, piú o meno in comune con tutti i dialetti Basic, ho trovato un articolo molto interessante sulla rivista Atari ST User che parla specificatamente di come accedere, tramite ST Basic, ad alcune funzioni dei componenti AES e VDI del sistema operativo.

### TOS, GEM, AES, VDI...

In informatica le sigle l'hanno sempre fatta da padrone e ovviamente l'Atari non difetta questo assioma. Il **TOS**, acronimo di **The Operating System**, é il sistema operativo alla base dei computer della serie ST. Apparso nel 1985 e sviluppato da Atari Corporation era un sistema operativo a finestre ed icone non equipaggiato di un multitasking reale, ma che grazie alla possibilitá di caricare in memoria dei programmi di tipo Terminate and Stay Resident (TSR), simulava un multitasking di tipo cooperativo.

Il **TOS**, rappresentato in figura 3, era costituito da questa serie di componenti:

- **Desktop** - L'interfaccia principale, caricata dopo l'avvio

- **GEM** - **Graphical Environment Manager**, composto a sua volta da:

- **AES** - **Application Environment System**, esegue operazioni sullo schermo, come la gestione delle finestre, il disegno e l'elaborazione delle finestre di dialogo, l'animazione dei menu e tante altre cose

- **VDI** - **Virtual Device Interface**, che gestisce funzioni di I/O grafico come disegnare linee ed eseguire operazioni di bit-blit

- **GEMDOS** - **GEM Disk Operating System** conosciuto anche come **BDOS** - **Basic Disk Operating System**, gestisce le interazioni con i dischi ed i file

- **BIOS** - **Basic Input/Output System**, gestisce le interazioni con l'hardware

- **XBIOS** - **Xtended BIOS**, gestisce chiamate a livello di hardware proprietarie dell'ST, come l'impostazione dei registri delle palette e la configurazione del chip audio.

Il VDI a sua volta si divide in tre componenti: font, driver e GDOS. Font definisce stili e dimensioni dei caratteri che possono essere scritti su un qualsiasi dispositivo di output (video, stampante...). Driver, come dice la parola stessa, é la parte di codice che implementa la macchina virtuale per un particolare dispositivo di output. Il GDOS é la parte device-independent del VDI, che (tra le altre cose) invia chiamate VDI al corretto dispositivo di output.

### VDISYS()

Il comando VDISYS() é soltanto brevemente accennato nel manuale dell'ST Basic. Peccato perché in realtà permette di accedere alla componente VDI ed utilizzare alcune funzionalità messe a disposizione dalla stessa.

Le funzioni possono essere invocate tramite il passaggio di alcuni parametri in tre variabili già definite: **CONTRL**, **PTSIN** e **INTIN** e richiamando poi la funzione VDISYS(1). Il valore 1 passato a VDISYS é puramente simbolico ed influente.

Le funzionalità che possono essere utilizzate sono: **BAR**, **CHARACTER HEIGHT**, **CHARACTER SLANT** e **TEXT TYPE**. Ma vediamo in dettaglio come fare.

### - Funzione BAR

La funzione **BAR** permette di disegnare un rettangolo pieno e può essere utilizzata per riempire o pulire intere aree dello schermo molto velocemente.

BAR - disegna un rettangolo pieno									
CONTRL				PTSIN				INTIN	
+0	+2	+6	+10	+0	+2	+4	+6	+10	
11	2	0	1	x start	y start	x end	y end		





Come si può intuire dalla tabella qui sopra, i parametri passati nella variabile CONTRL sono fissi e servono per invocare la funzione BAR, mentre in PTSIN sono indicate le coordinate di inizio e fine della diagonale del rettangolo. Il programma di esempio chiarirá definitivamente tutti i dubbi.

```
10 fullw 2:clearw 2:color 1,1,1,1,1
20 poke contrl,11
30 poke contrl+2,2
35 poke contrl+4,0
40 poke contrl+10,1
50 poke ptsin,50
60 poke ptsin+2,50
70 poke ptsin+4,150
80 poke ptsin+6,150
90 vdisys(1)
100 color 1,1,1,2,24
110 poke ptsin+4,120
120 poke ptsin+6,120
130 vdisys(1)
140 color 1,1,1,1,1
150 poke ptsin+4,80
160 poke ptsin+6,80
170 vdisys(1)
```

### - Funzione CHARACTER HEIGHT

Come facilmente intuibile dal nome, questa funzione permette di gestire la dimensione del carattere. Vediamo i parametri che utilizza.

CHARACTER HEIGHT - definisce la dimensione del carattere									
CONTRL				PTSIN					INTIN
+0	+2	+6	+10	+0	+2	+4	+6	+10	
12	1	0		0	Size				

La dimensione del carattere é specificata in PTSIN+2. IL default é 13.

### - Funzione CHARACTER SLANT

Questa funzione permette di specificare l'orientamento del testo.

CHARACTER SLANT - definisce l'orientamento dei caratteri									
CONTRL				PTSIN					INTIN
+0	+2	+6	+10	+0	+2	+4	+6	+10	
13	0	1							Angle

L'orientamento é specificato nella variabile INTIN ed é espresso in decimi di grado. Sebbene sia possibile specificare valori da 0 a 3600, solo i valori 0, 900, 1800 e 3600 produrranno un risultato a video.

### - Funzione TEXT TYPE

Questa funzione permette di specificare lo stile del testo.

TEXT TYPE - definisce lo stile del testo									
CONTRL				PTSIN					INTIN
+0	+2	+6	+10	+0	+2	+4	+6	+10	
106	0	1							VALUE

Lo stile del testo é definito dal valore specificato nella variabile INTIN in questo modo:

- 0 - normale
- 1 - bold - grassetto
- 2 - grey - grigio
- 4 - italics - corsivo
- 8 - underlined - sottolineato
- 16 - outlined - delineato
- 32 - shadow - ombreggiato

Vediamo un esempio pratico che dovrebbe chiarirne l'utilizzo:

```
10 fullw 2:clearw 2:color 1,1,1,1,1
20 poke contrl,106
30 poke contrl+2,0
40 poke contrl+6,1
50 poke intin,8
55 vdisys(1)
60 print "Francesco"
70 poke intin,4
75 vdisys(1)
80 print "Francesco"
85 poke intin,0
90 vdisys(1)
```

Ovviamente gli effetti possono essere combinati semplicemente sommando i valori. Per ottenere un corsivo grassetto, per esempio, basterá passare il valore 5 ad INTIN. Ah, ricordatevi di settare di nuovo il valore 0 altrimenti anche il testo dell'ST Basic continuerá ad utilizzare lo stile che avete impostato.

### GEMSYS()

Il comando per invocare le funzionalitá dell'AES é il GEMSYS. Proprio come il comando VDISYS, GEMSYS richiede dei parametri per specificare le funzionalitá da attivare. Questi parametri sono passati all'AES in array chiamati **CONTRL**, **INTIN**, **INTOUT**, **ADDRIN**, **ADDROUT** e **GLOBAL**. Beh, che problema c'è, conosciamo già alcune di queste strutture dati... Eh no, non é cosí semplice. Queste strutture dati non sono le stesse che abbiamo utilizzato con il VDISYS. Sono strutture dati diverse che risiedono anche in aree diverse della memoria.

Come fare per accedere a queste strutture dati visto che le variabili che abbiamo utilizzato con VDISYS non





Returns the current mouse location, mouse button state and keyboard state.

BASIC CODE	DESCRIPTION	BASIC CODE	DESCRIPTION
1 a#=gb		7 kb=peek(gintout+8)	kb= Current keyboard state
2 gintout=peek(a#+12)	Define integer output	0	No key pressed
3 gemsys(79)	OPCODE	1	Right shift key pressed
4 mx=peek(gintout+2)	mx=Coordinate of mouse's current location	2	Left shift key pressed
5 my=peek(gintout+4)	my=Coordinate of mouse's current location	4	Control key pressed
6 mb=peek(gintout+6)	mb=Current mouse button state.	8	Alternate key pressed
	0 No button pressed		
	1 Left button pressed		
	2 Right button pressed		

Fig. 4 - AES\_OPCODE 73 - Mouse status

funzionano? L'ST Basic mette a disposizione una variabile di sistema chiamata GB che é un puntatore come gli altri, che però non punta ad un campo di dati, ma ad una tabella dove sono memorizzati altri puntatori. E questi sono i puntatori di cui abbiamo bisogno. Per trovare i puntatori alle strutture che ci interessano possiamo usare la seguente tabella:

VARIABILE	PUNTATORE
CONTRL	PEEK(GB)
GLOBAL	PEEK(GB+4)
INTIN	PEEK(GB+8)
INTOUT	PEEK(GB+12)
ADDRIN	PEEK(GB+16)
ADDROUT	PEEK(GB+20)

A differenza del VDISYS però la funzione GEMSYS passa l'OPCODE nella chiamata di funzione.

Si dia un'occhiata al seguente programma:

```

10 b#=gb
15 io=peek(b+12)
20 cn=peek(b)
30 poke cn+8,0
40 poke cn+16,0
50 gemsys(79)
60 mx=peek(io+2)
62 mx=peek(io+4)
64 print "POS x= ";mx
65 print "POS y= ";my

```

Per comprendere il programma si consultino le informazioni presenti in Figura 4.

**Attenzione:** Stavo veramente impazzendo per capire come mai, nonostante il listato presentato fosse corretto, ricevevo sempre e comunque x=0 e y=252 come risposta.

Poi ho trovato questo articolo: <http://gfabasic.net/stg/>

RETROMAGAZINE WORLD ANNO 6 - NUMERO 36

*gfabasic.htm#GRAF\_MKSTATE* dove, a proposito della funzione *GRAF\_MKSTATE*, si dice "This is an AES routine to query the mouse. Unlike *MOUSEX* etc., the function gives valid results if the pointer is within a menu bar."

Per una lista completa delle funzionalità offerte da AES si veda: <https://www.seasip.info/Gem/aes.html>

Per il momento mi fermo qui, ben conscio di aver soltanto sfiorato la cima dell'iceberg che si cela dietro l'ST Basic, l'Atari ST ed il suo sistema operativo TOS.

Buon divertimento!

#### Link utili

Riporto qui i link ad alcuni documenti che ho trovato utili per scrivere l'articolo.

- <https://www.atarimagazines.com/startv1n1/TrackingElusiveGDOS.html>

- [http://www.atarimania.com/atari-magazine-atari-st-user\\_17.html](http://www.atarimania.com/atari-magazine-atari-st-user_17.html)

- <https://docs.dev-docs.org/>

- [https://archive.org/details/Understanding\\_Atari\\_ST\\_BASIC\\_Programming/page/n165/mode/2up](https://archive.org/details/Understanding_Atari_ST_BASIC_Programming/page/n165/mode/2up)





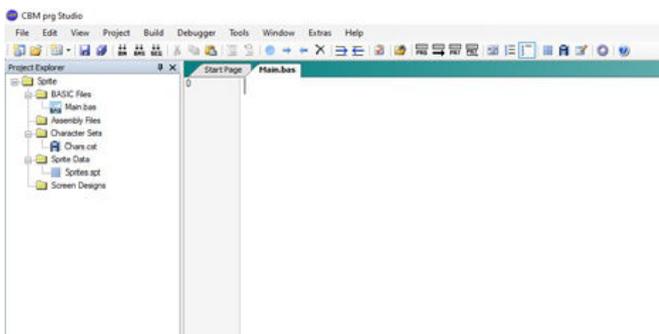
# Come creare un gioco in BASIC per Commodore 64 - parte 2

di Felice Nardella (aka Kimono)

## GLI SPRITES (parte #1)

Quello degli Sprites è un argomento di fondamentale importanza per i videogiochi; si può dire che gli Sprites (oltre al SID) abbiano contribuito non poco alla “fortuna” che il C64 ebbe negli anni '80, e che ancora oggi, a distanza di quasi 40 anni(!), gli donino un fascino del tutto particolare, grazie al quale il notevole interesse che ruota intorno a questa straordinaria “macchina d'epoca”, nonostante l'età, è ancora vivo tra noi, inguaribili nostalgici.

Ma procediamo, ora, senza troppi fronzoli, con una breve descrizione delle caratteristiche di base degli sprites: innanzitutto diciamo che, in Basic, il C64 è in grado di gestire e visualizzare al massimo 8 sprites contemporaneamente, che vengono numerati da 0 a 7. Orbene, per prendere subito familiarità con gli sprites, avviate il CBM prg Studio e create un nuovo Progetto Basic (File → New BASIC Project) chiamato “Sprite”, avendo cura di lasciare la spunta a “Add a Sprite Data file called” e rinominate il file “Main.bas” in “Mongolfiera.bas”.



Poi cliccate 2 volte su “Sprites.spt” e verrà fuori la schermata di creazione degli sprites; dopodiché andate su “Sprites” → “New”; se avete eseguito tutto correttamente, vedrete una finestra come quella in figura 1.

Come potete intuire, all'interno di quella griglia dovremo andare a disegnare il nostro sprite, annerendone i quadratini. La griglia è formata da 21 righe ed ogni riga è formata da 24 quadratini: ogni quadratino rappresenta 1 bit. Poiché 8 bit formano 1 byte, ogni riga è formata da 3 bytes. In totale, dunque, ogni sprite occupa  $21 * 3 = 63$  bytes. In realtà ne occupa 64, poiché alla fine di ogni sprite, ci deve essere un byte nullo. Riepilogando, ogni sprite occupa in memoria 64 bytes di

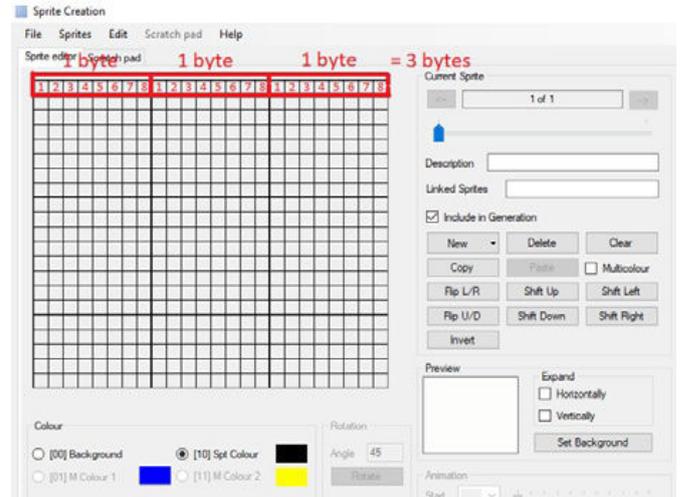
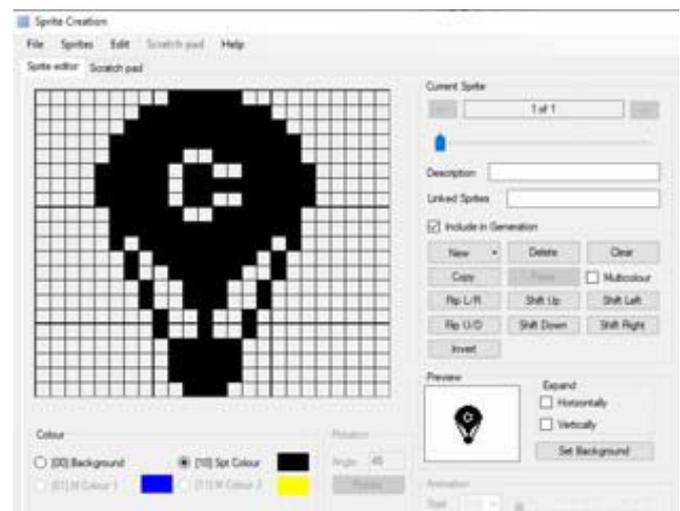


Fig. 1 - Sprite Editor di CBM Prg Studio

RAM ed è definito da 24 punti in larghezza e 21 punti in altezza, per un totale di 504 punti ( $24 * 21$ ).

## LET'S START!

Cominciamo a disegnare il nostro sprite sulla griglia: chi di voi non ricorda la famosa mongolfiera presentata nel classico manuale in dotazione al C64? Ebbene, ora è arrivato il momento di riprodurla sul CBM prg Studio! Cercate di disegnare una mongolfiera, tipo quella che vedete in figura:



Poi salvatela cliccando su “File” → “Save”, quindi andate a creare i DATA del vostro sprite cliccando su “File” → “Export” → “To Listing...” e vi si aprirà una finestra del genere:





Sprite Data Generation

Generate as  
 BASIC Data  
 Assembly Data

Line Numbering  
 Start Line Numbers At:   
 Increment:

Data  
 Generate data using  values per line  
 Generate using '@' for 'ON' character and '.' for 'OFF' character  
 Pad Data to 64 Bytes  Use description as label  
 Include Sprite number in description

Format  
 Decimal  Hexadecimal

DATA and REM Statements  
 Upper case  Lower case

Generation Control  
 Generate data for all Sprites  
 Only Generate Included Sprites  
 Specify range

Settate tutto come in figura e premete OK. In tal modo il "CBM prg Studio" avrà generato per voi il seguente listato con i DATA dello sprite:

```
100 rem
110 data 0,124,0,1,255,0,3,255,128,7,255,192,
7,207,192,15,179,224,15,191
120 data 224,15,179,224,7,207,192,7,255,192,5,
255,64,2,254,128,2,124,128,1
130 data 57,0,1,57,0,0,146,0,0,146,0,0,124,0,
0,124,0,0,124,0
140 data 0,56,0,0
```

Come potete forse intuire, negli anni '80 gli sprites venivano disegnati a mano ed i valori dei dati del disegno, che CBM prg Studio ha calcolato per noi, venivano anch'essi calcolati a mano!

Adesso che abbiamo ricavato i dati del disegno, grazie al CBM prg Studio, dobbiamo cercare di immagazzinarli da qualche parte nella RAM: occorre in definitiva, cercare uno spazio di almeno 63 bytes contigui; inoltre tali blocchi di bytes devono cominciare da locazioni che siano multiple di 64.

Per farla breve, per ora potete prendere per buone le seguenti locazioni, da cui partire per immagazzinare i dati di uno sprite:

- 11 \* 64 = 704
- 13 \* 64 = 832
- 14 \* 64 = 896
- 15 \* 64 = 960

Ad esempio, usiamo le locazioni a partire dalla 704,

scrivendo la seguente riga codice:

```
10 for n=704 to 704+63: read q:poke n,q: next
```

## PUNTATORI AGLI SPRITE

Una volta memorizzati i dati dello sprite, occorre far sapere al C64 dove li abbiamo memorizzati, altrimenti da solo non potrà mai riuscire a trovarli.

Per far ciò ci serviremo dei puntatori agli sprite, che si trovano in un blocco di 8 bytes a partire dalla locazione 2040 fino alla 2047. La locazione per il puntatore allo sprite 0 è la 2040, quella per il puntatore allo sprite 1 è la 2041, e così via per tutti e 8 gli sprite.

SPRITE	PUNTATORI AGLI SPRITE
#0	2040
#1	2041
#2	2042
#3	2043
#4	2044
#5	2045
#6	2046
#7	2047

Poiché abbiamo un solo sprite da visualizzare, al momento ci serviremo solo della locazione 2040 che dovrà riferirsi alla locazione 704.

Pertanto, dovremo aggiungere al nostro programma la riga seguente:

```
20 poke 2040,11
```

Ciò in quanto  $704 / 64 = 11$ .

Se invece, lo sprite 0 fosse stato, ad esempio, immagazzinato a partire dalla locazione 832, avremmo dovuto scrivere

```
20 poke 2040,13
```

Poiché  $832 / 64 = 13$ .

## MAPPA DEI REGISTRI

Orbene, per poter proseguire col nostro programma dovremo soffermarci un attimo sul chip che gestisce la grafica del C64: il VICII.

Ci serviranno soprattutto i suoi registri, che partono dal byte 53248; fate riferimento alla mappa dei registri del VIC- II in Figura 2.

Analizziamo i registri che, per il momento, serviranno al nostro obiettivo di visualizzare uno sprite:

### a) Registri 53248, 53249 (Coordinate Sprite 0)

Come potete osservare sulla mappa, i registri 53248 e 53249 definiscono rispettivamente le coordinate x e y dello sprite 0.

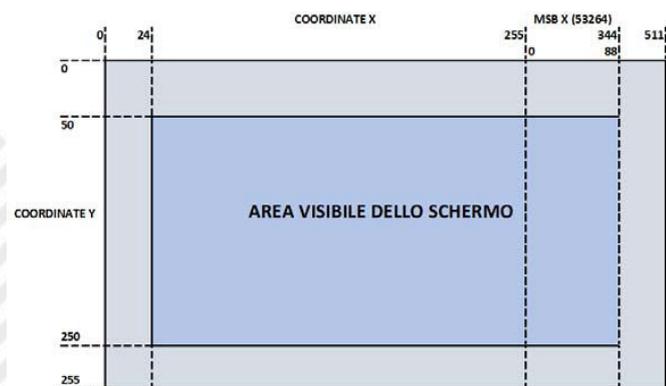




Dec Address	Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Contents
53248	Register				M0X					X Coordinate Sprite 0
53249	Register				M0Y					Y Coordinate Sprite 0
53250	Register				M1X					X Coordinate Sprite 1
53251	Register				M1Y					Y Coordinate Sprite 1
53252	Register				M2X					X Coordinate Sprite 2
53253	Register				M2Y					Y Coordinate Sprite 2
53254	Register				M3X					X Coordinate Sprite 3
53255	Register				M3Y					Y Coordinate Sprite 3
53256	Register				M4X					X Coordinate Sprite 4
53257	Register				M4Y					Y Coordinate Sprite 4
53258	Register				M5X					X Coordinate Sprite 5
53259	Register				M5Y					Y Coordinate Sprite 5
53260	Register				M6X					X Coordinate Sprite 6
53261	Register				M6Y					Y Coordinate Sprite 6
53262	Register				M7X					X Coordinate Sprite 7
53263	Register				M7Y					Y Coordinate Sprite 7
53264	Register	M7X8	M6X8	M5X8	M4X8	M3X8	M2X8	M1X8	M0X8	MSBs of X coordinates
53265	Register	RST8	ECM	BMM	DEN	RSEL		YSCROLL		Control register 1
53266	Register					RASTER				Raster counter
53267	Register					LPX				Light pen X
53268	Register					LPY				Light pen Y
53269	Register	M7E	M6E	M5E	M4E	M3E	M2E	M1E	M0E	Sprite enabled
53270	Register	-	-	RES	MCM	CSEL		XSCROLL		Control register 2
53271	Register	M7YE	M6YE	M5YE	M4YE	M3YE	M2YE	M1YE	M0YE	Sprite Y expansion
53272	Register	VM13	VM12	VM11	VM10	CB13	CB12	CB11	-	Memory pointers
53273	Register	IRQ	-	-	-	ILP	IMMC	IMBC	IRST	Interrupt register
53274	Register	IRQ	-	-	-	ELP	EMMC	EMBC	ERST	Interrupt enabled
53275	Register	M7DP	M6DP	M5DP	M4DP	M3DP	M2DP	M1DP	M0DP	Sprite data priority
53276	Register	M7MC	M6MC	M5MC	M4MC	M3MC	M2MC	M1MC	M0MC	Sprite multicolour
53277	Register	M7XE	M6XE	M5XE	M4XE	M3XE	M2XE	M1XE	M0XE	Sprite X expansion
53278	Register	M7M	M6M	M5M	M4M	M3M	M2M	M1M	M0M	Sprite-sprite collision
53279	Register	M7D	M6D	M5D	M4D	M3D	M2D	M1D	M0D	Sprite-data collision
53280	Register	-	-	-	-			EC		Border colour
53281	Register	-	-	-	-			B0C		Background colour 0
53282	Register	-	-	-	-			B1C		Background colour 1
53283	Register	-	-	-	-			B2C		Background colour 2
53284	Register	-	-	-	-			B3C		Background colour 3
53285	Register	-	-	-	-			MM0		Sprite multicolour 0
53286	Register	-	-	-	-			MM1		Sprite multicolour 1
53287	Register	-	-	-	-			M0C		Sprite 0 colour
53288	Register	-	-	-	-			M1C		Sprite 1 colour
53289	Register	-	-	-	-			M2C		Sprite 2 colour
53290	Register	-	-	-	-			M3C		Sprite 3 colour
53291	Register	-	-	-	-			M4C		Sprite 4 colour
53292	Register	-	-	-	-			M5C		Sprite 5 colour
53293	Register	-	-	-	-			M6C		Sprite 6 colour
53294	Register	-	-	-	-			M7C		Sprite 7 colour

Fig. 2 - Mappa dei registri del VIC- II

Allo scopo di visualizzare tali coordinate sullo schermo, ho preparato la seguente figura che vedete qui sotto.



Come si può notare, le coordinate hanno origine nell'angolo superiore sinistro. Per quanto riguarda le coordinate Y, esse vanno da 0 a 255, ma quelle che fanno parte dell'area visibile dello schermo vanno da 50 a 250.

Qualche parola in più, invece, occorre spendere per le coordinate X, che vanno da 0 a 511. Come mostrato in figura, quelle che ricadono nell'area visibile dello schermo,

partono da 24 e vanno ben oltre 255, ma nonostante tutto, non è possibile andare oltre la coordinata 255, a meno che non si vada ad agire anche sul registro 53264, detto registro dei bit più significativi (o MSB = Most Significant Bit).

Ciò vuol dire che se volessimo posizionare lo sprite 0 alla coordinata  $x = 256$ , non potremo scrivere POKE 53240,256 (no!), bensì dovremo scrivere:

**POKE 53264,1 : POKE 53240,0**

Poiché vogliamo che la nostra mongolfiera parta dall'angolo inferiore sinistro dello schermo aggiungeremo la seguente riga:

**40 poke 53248,24 : poke 53249,229**

Da notare che il valore 229 è dato dalla coordinata max di Y (250) da cui è stata sottratta l'altezza dello sprite (21). Infatti  $250 - 21 = 229$ .

### b) Registro 53269 (abilita gli sprites)

Un'altra locazione importante per i nostri scopi è la 53269 che serve ad abilitare gli sprites, cioè a renderli visibili, oppure, al contrario a disattivarli: ad esempio, per abilitare lo sprite 0 occorrerà aggiungere la seguente riga al programma:

**30 poke 53269,1**

Ciò in quanto  $20 = 1$ .

Se invece, avessimo dovuto attivare lo sprite 1, avremmo dovuto scrivere:

**30 poke 53269,2**

Poiché  $21 = 2$ .

Se ancora, avessimo dovuto attivare sia lo sprite 0 che lo sprite 1, avremmo dovuto scrivere:

**30 poke 53269,3**

Poiché  $20 + 21 = 3$

Parimenti, per poter disattivare uno sprite, occorre porre a 0 il bit ad esso corrispondente; ad esempio, per spegnere tutti gli sprites (e quindi farli scomparire), occorre scrivere: **poke 53269,0**

In tal modo tutti e 8 i bit saranno posti a 0.

### c) Registro 53287 (colore dello sprite 0)

Questo registro indica il colore che avrà il nostro sprite 0. In questo caso scegliamo il ciano il cui codice, come sappiamo, è 3; per cui occorre aggiungere al programma anche la seguente linea:

**50 poke 53287,3**

### VISUALIZZARE LO SPRITE

Finalmente, ora che abbiamo esaminato tutto ciò che

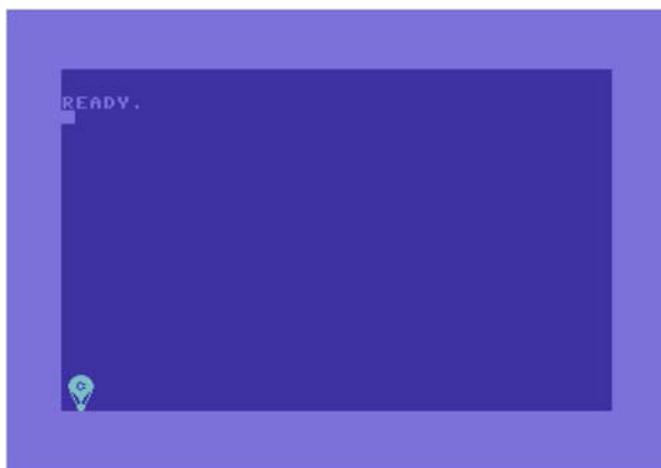




serve per creare e visualizzare uno sprite, possiamo digitare il nostro programma, semplicemente riepilogando quanto descritto finora; quindi il nostro programmino si presenterà come segue:

```
0 rem pulisce lo schermo
1 print"{clear}"
5 rem carica in memoria i dati dello sprite
10 for n=704 to 704+63: read q:poke n,q: next
15 rem puntatore allo sprite 0
20 poke 2040,11
25 rem attiva lo sprite 0
30 poke 53269,1
35 rem coordinate x, y dello sprite 0
40 poke 53248,24: poke 53249,229
45 rem colore dello sprite 0 (3 = ciano)
50 poke 53287,3
100 rem dati dello sprite 0
110 data 0,124,0,1,255,0,3,255,128,7,255,192,
7,207,192,15,179,224,15,191
120 data 224,15,179,224,7,207,192,7,255,192,5,
255,64,2,254,128,2,124,128,1
130 data 57,0,1,57,0,0,146,0,0,146,0,0,124,0,
0,124,0,0,124,0
140 data 0,56,0,0
```

Ora salviamolo, compiliamolo e mandiamolo in esecuzione sull'emulatore; ecco come si presenterà lo schermo:



### MOVE YOUR SPRITE!!

Certo, avere un'immagine statica non è il massimo della vita: cerchiamo, quindi, di dargli un po' di movimento! Come faremo?

Semplicemente operando sulle coordinate X,Y dello sprite. Vogliamo farlo muovere dal basso verso l'alto e da sinistra verso destra, quindi dobbiamo incrementare il valore della coordinata X e contemporaneamente diminuire quello della coordinata Y. Uno dei modi per far ciò, è il seguente:

```
65 rem incrementa la coord. x e decrementa
la coord. y
70 for x=24 to 255 step 2: y=y-1
75 rem coordinate x, y dello sprite 0
80 poke 53248,x: poke 53249,y
85 next
```

Quindi il programma, nella sua interezza, diventerà:

```
5 rem pulisce lo schermo
10 print"{clear}"
15 rem carica in memoria i dati dello sprite
20 for n=704 to 704+63: read q:poke n,q: next
25 rem puntatore allo sprite 0
30 poke 2040,11
35 rem attiva lo sprite 0
40 poke 53269,1
45 rem colore dello sprite 0 (3 = ciano)
50 poke 53287,3
55 rem imposta la coordinata y iniziale
60 y=229
65 rem incrementa la coord. x e decrementa
la coord. y
70 for x=24 to 255 step 2: y=y-1
75 rem coordinate x, y dello sprite 0
80 poke 53248,x: poke 53249,y
85 next
90 rem ripete il ciclo dalla riga 60
95 goto 60
100 rem dati dello sprite 0
105 data 0,124,0,1,255,0,3,255,128,7,255,192,
7,207,192,15,179,224,15,191
110 data 224,15,179,224,7,207,192,7,255,192,5,
255,64,2,254,128,2,124,128,1
115 data 57,0,1,57,0,0,146,0,0,146,0,0,124,
0,0,124,0,0,124,0
120 data 0,56,0,0
```

Adesso vogliamo invece, che la mongolfiera riesca superare la "barriera dei 255" sull'asse x, per cui agendo sul registro MSB, potremo far svolazzare la nostra mongolfiera per tutta la lunghezza dello schermo visibile. Ecco come si presenterà il nostro programma:

```
5 rem pulisce lo schermo
10 print"{clear}"
15 rem carica in memoria i dati dello sprite
20 for n=704 to 704+63: read q:poke n,q: next
25 rem puntatore allo sprite 0
30 poke 2040,11
```





```

35 rem attiva lo sprite 0
40 poke 53269,1
45 rem colore dello sprite 0 (3 = ciano)
50 poke 53287,3
55 rem imposta la coordinata y iniziale;
  azzerà i bit nel registro MSB
60 y=229:poke 53264,0
65 rem incrementa la coord. x e decrementa
  la coord. y
70 for x=24 to 255 step 2: y=y-1
75 rem coordinate x, y dello sprite 0
80 poke 53248,x: poke 53249,y
85 next
90 rem abilita il bit per lo sprite 0 nel
  registro MSB
95 poke 53264,1
100 rem incrementa la coord. x e decrementa
  la coord. y
105 for x=0 to 68 step 2: y=y-1
110 rem coordinate x, y dello sprite 0
115 poke 53248,x: poke 53249,y
120 next
125 rem ripete il ciclo dalla riga 60
130 goto 60
135 rem dati dello sprite 0
140 data 0,124,0,1,255,0,3,255,128,7,255,192,
  7,207,192,15,179,224,15,191
145 data 224,15,179,224,7,207,192,7,255,192,
  5,255,64,2,254,128,2,124,128,1
150 data 57,0,1,57,0,0,146,0,0,146,0,0,124,
  0,0,124,0,0,124,0
155 data 0,56,0,0

```

## VISUALIZZARE PIÙ SPRITES

Per poter visualizzare più sprites contemporaneamente occorre semplicemente rammentare tutto ciò che è stato descritto finora, tenendo sempre presente la mappa dei registri. Occorrerà quindi:

- 1) Aggiungere tanti puntatori agli sprites (da 2040 a 2047), quanti sono gli sprites che vogliamo visualizzare (max 8);
- 2) Ricordarsi di attivarli (registro 53269);
- 3) Assegnare ad ognuno una posizione in termini di coordinate X,Y (registri da 53248 a 53263);
- 4) Assegnare ad ognuno un colore (registri da 53287 a 53294).

Seguendo questi 4 semplici punti, saremo in grado di rappresentare più sprites assieme.

Naturalmente se nei puntatori agli sprite inseriremo lo stesso indirizzo (nel nostro caso 11), andremo a visualizzare 2 o più sprites identici (magari di colore diverso, ma comunque di forma uguale).

Se invece, volessimo rappresentare 2 o più sprites diversi, occorrerà prima inserire in RAM le diverse forme date agli sprites, attraverso i DATA, e poi inserire gli indirizzi corrispondenti ai diversi sprites nei vari puntatori.

Torneremo su questo aspetto nel prossimo capitolo. Per il momento darò un paio di esempi di visualizzazione di più sprites (uguali) contemporaneamente: il primo visualizza due mongolfiere che ondeggiando insieme, una in alto e l'altra in basso, con l'ausilio delle funzioni trigonometriche sin() e cos().

Ecco il listato completo e commentato del primo esempio appena descritto:

```

5 rem pulisce lo schermo
10 print"{clear}"
15 rem carica in memoria i dati dello sprite
20 for n=704 to 704+63: read q:poke n,q: next
25 rem puntatore agli sprite 0 e 1
30 poke 2040,11:poke 2041,11
35 rem attiva gli sprite 0 e 1
40 poke 53269,7
45 rem colore degli sprite 0 e 1 (3 = ciano,
  2 = rosso)
50 poke 53287,3: poke 53288,2
55 forx=24to255
60 rem calcola le traiettorie verticali
65 y=40*sin(x/2)+100:y1=40*cos(x/2)+180
70 rem coordinate x,y degli sprite 0 e 1
75 poke 53248,x: poke 53249,y
80 poke 53250,x: poke 53251,y1
85 next
90 rem abilita il bit per gli sprite 0 e 1
  nel registro MSB
95 poke 53264,3
100 rem coordinate x,y degli sprite 0 e 1
105 forx=0to68
110 rem calcola le traiettorie verticali
115 y=40*sin(x/2)+100:y1=40*cos(x/2)+180
120 poke 53248,x: poke 53249,y
125 poke 53250,x: poke 53251,y1
130 next
135 rem dati dello sprite 0
140 data 0,124,0,1,255,0,3,255,128,7,255,192,
  7,207,192,15,179,224,15,191

```



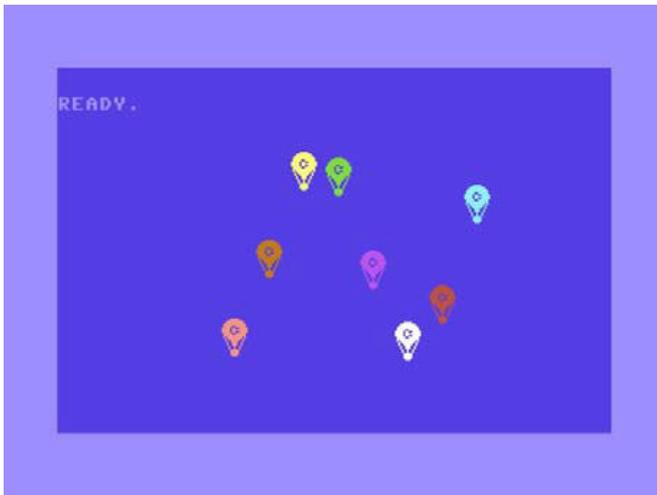


```

145 data 224,15,179,224,7,207,192,7,255,192,
5,255,64,2,254,128,2,124,128,1
150 data 57,0,1,57,0,0,146,0,0,146,0,0,124,
0,0,124,0,0,124,0
155 data 0,56,0,0

```

Il prossimo esempio visualizza ben 8 mongolfiere di diverso colore, che ondeggiando in fila, una dietro l'altra sullo schermo, tracciando una traiettoria sinusoidale.



Ecco il listato completo:

```

5 rem pulisce lo schermo
10 print"{clear}"
15 rem carica in memoria i dati dello sprite
20 for n=704 to 704+63: read q:poke n,q: next
25 rem puntatore agli sprite
30 poke 2040,11:poke 2041,11:poke2042,11:poke
2043,11
35 poke 2044,11:poke 2045,11:poke2046,11:poke
2047,11
40 rem attiva tutti e 8 gli sprite
45 poke 53269,255
50 rem colore degli sprite
55 poke 53287,3: poke53288,2:poke 53289,1:poke
53290,4
60 poke 53291,5:poke 53292,7:poke53293,8:poke
53294,10
65 rem definisce la funzione f(a)
70 def fn f(a)=50*sin(a)+140
75 rem coordinate x, y degli sprite
80 forx=140to255
85 poke 53248,x: poke 53249,fn f(x)
90 poke 53250,x-20: poke 53251,fn f(x-20)
95 poke 53252,x-40: poke 53253,fn f(x-40)
100 poke 53254,x-60: poke 53255,fn f(x-60)
105 poke 53256,x-80: poke 53257,fn f(x-80)
110 poke 53258,x-100: poke 53259,fn f(x-100)

```

Appuntamento al prossimo numero con la terza puntata del tutorial!

*Tutta la redazione di RetroMagazine World ringrazia ed apprezza il lavoro del gruppo Facebook "RetroProgramming Italia - RP Italia" - A division of "RetroCampus" Associazione Culturale e li ringrazia e per la gentile concessione dei loro contributi.*

*Riteniamo che, a pari di altri gruppi simili, stiano facendo un ottimo lavoro per far conoscere il panorama del retrocoding italiano nel mondo.*

*Invitiamo i nostri lettori a collaborare con loro nel caso avessero materiale da condividere con tutti gli appassionati di retrocomputing e retrocoding.*





# C128 in modalità 80 colonne - parte 3

di Gianluca Girelli

Come menzionato negli articoli precedenti, il chip 8563 VDC è sicuramente un componente molto potente e versatile del C128. Abbiamo imparato che, a differenza del VIC, i suoi registri non possono essere indirizzati direttamente usando i comandi POKE/PEEK, anche se tali comandi possono essere impiegati per interagire con i registri di I/O. La mancanza di questa caratteristica rende l'interazione con il chip un po' fastidiosa, ma i risultati che si possono ottenere con un pizzico di fatica in più sono eccezionali e gratificanti.

Per darvi un assaggio di ciò che è effettivamente possibile, ho deciso di incentrare il presente articolo su una serie di brevi programmi che mostrano alcune di queste possibilità. Spero che questi frammenti di codice, presi da un vecchio libro "Data Becker" pubblicato da "Abacus Software" nel 1985, vi facciano venire voglia di ulteriori e più approfonditi test.

Senza ulteriori indugi, iniziamo con un programma che mostra come si può ottenere uno "smooth scroll" del video. Come con il chip VIC, è possibile spostare lo schermo verticalmente o orizzontalmente in incrementi di linee raster sul VDC; questo è fatto usando i registri VDC 24 e 25 ("VERTICAL SMOOTH SCROLL" e "HORIZONTAL SMOOTH SCROLLING" rispettivamente) dove i suoi bit 0-4 (per il registro 24) e 0-3 (per il registro 25) sono utilizzati per questo scopo. Contrariamente al modo in cui lo smooth scrolling è fatto sul VIC, sul VDC non si perde nessuna colonna o linea.

E' bene sottolineare che il VDC non è adatto ai giochi: nonostante abbia una risoluzione molto buona, il suo complicato indirizzamento è troppo lento per essere utile in un gioco, ma si può usare lo smooth scrolling per creare effetti visivi interessanti.

Si noti che se il codice "gira" in modo troppo veloce (o non abbastanza velocemente), è sufficiente cambiare i cicli di ritardo nelle linee 130 e 160.

Se il bit 3 è azzerato, vengono visualizzate 25 linee e la seguente (o precedente) sezione della RAM viene fatta scorrere sullo schermo (ricordate dagli articoli precedenti che il VDC ha una propria memoria indipendente). Se si imposta il bit 3, vengono visualizzate solo 22 linee e potete far scorrere le ultime tre linee dello schermo per mezzo dello smooth scrolling.

```

10 rem *** demo program for smooth scrolling ***
20 a=dec("d600"):d=dec("d601")
30 ve=24:ho=25
40 print chr$(147)chr$(27);"m";: rem screen clr
and scroll off
50 a$="hello c-128 fans!"
60 for i=0 to 24
70 print a$
80 next
90 :
100 for i0=0 to 6
110 poke a,ve:v=peek(d) and 240 or i0
120 poke a,ve: poke d,v
130 for i1=1 to 20: next
140 poke a,ho: h=peek(d) and 240 or i0
150 poke a,ho: poke d,h
160 for i1=1 to 20: next
170 next
180 goto 100

```

Come potete vedere dalla Figura 1, lo schermo mostra linee corrotte nella parte bassa del video; arriveremo a spiegare il perchè tra un po'. Per il momento, concentriamoci su un altro punto: abbiamo appena detto che questa funzione non è abbastanza veloce per i giochi, tuttavia, se il controller è così difficile da manipolare, perché lo scorrimento dello schermo è così veloce?

La soluzione è semplice: il VDC è abbastanza "intelligente" da spostare interi blocchi da una parte all'altra della sua memoria. Se questo dovesse essere fatto tramite l'indirizzamento relativo, ci vorrebbe un tempo considerevolmente più lungo. Invece, se volete che il VDC sposti un'area di memoria, basta dirglielo tramite il bit

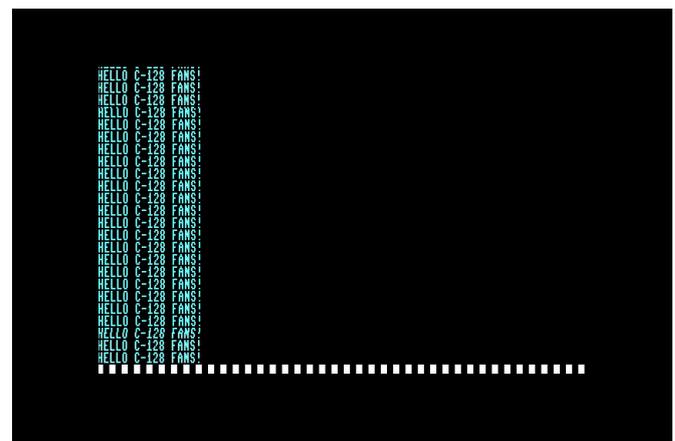


Fig. 1



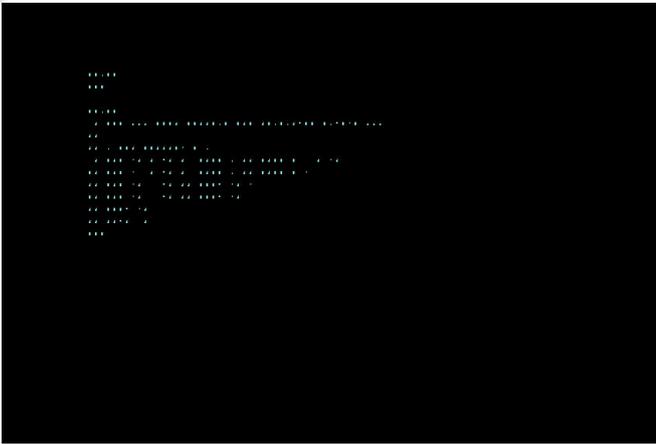


Fig. 2

COPY (bit 7 in REG 24). Se questo bit è impostato, il VDC copia invece di riempire. L'indirizzo di partenza del blocco da copiare è definito nei registri 32 e 33; l'indirizzo di destinazione della procedura di copia deve essere definito nel registro di aggiornamento (REG 18 e 19); il processo di copia inizia quando si scrive nel registro di "word count". Questo specifica anche il numero di caratteri da copiare.

**NOTA:** Il registro "word count" specifica il numero esatto di caratteri da copiare. Per esempio, se volete copiare la prima riga di testo dello schermo sulla riga sottostante e preservare gli attributi, dovete prima copiare la linea di testo e poi gli attributi. Nel nostro esempio realizzeremo invece uno scorrimento verso l'alto; in BASIC il codice girerebbe abbastanza lentamente, ma in linguaggio macchina la velocità è accettabile.

In questo caso particolare un'immagine non aiuterebbe ma, se eseguite il codice, noterete che l'intero schermo si muove verso l'alto per mezzo della copia a blocchi.

```
10 rem *** demo program for block copying ***
20 a=dec("d600"):d=dec("d601")
30 poke a,24: c=peek(d):rem *** contents of reg 24
40 poke a,24: poke d,c or 128:rem *** set copy bit
50 for z=24 to 0 step -1
60 aq=z*80: az=aq+80: rem * * * source and dest
70 poke a,18:poke d,az/256:poke a,19:poke d,az
and 255
80 poke a,32:poke d,aq/256:poke a,33:poke d,aq
and 255
90 poke a,30:poke d,79: rem *** copy text
100 aq=2048+aq:az=2048+az:rem *** attribute addr
110 poke a,18:poke d,az/256:poke a,19: poke d,az
and 255
120 poke a,32:poke d,aq/256:poke a,33: poke d,aq
and 255
130 poke a,30: poke d,79: rem *** copy attribute
```



Fig. 3

```
140 next
150 print chr$(19);chr$(27)"d"; :rem clear 1st line
160 poke a,24:poke d,c: rem *** clear copy bit
```

Esploriamo ora come la matrice dei caratteri può essere alterata. Come sapete, in modo analogo al chip VIC, anche la matrice dei caratteri che si trova nella RAM VDC è di 8x8 punti; questo significa che i caratteri visualizzati sullo schermo sono larghi 8 punti e alti altrettanto. Questo può essere cambiato armeggiando con i registri 22 e 23, "CHARACTER TOTAL & DISPLAYED" e "CHARACTER DSP(V)" rispettivamente, dove il primo determina il numero totale di linee orizzontali visualizzate (e la larghezza del carattere) mentre il secondo determina il numero di linee verticali (e l'altezza di un carattere). Le figure da 2 a 4 mostrano il codice mentre viene eseguito.

```
10 rem *** demo program for character matrix ***
20 :
30 a=dec("d600"):d=a+1
40 for i0=0 to 8: poke a,22:poke d,112+i0
50 for i1=0 to 8: poke a,23:poke d,i1
60 for i2=1 to 30:next i2,i1
70 for i2=1 to 30:next i2
80 next i0
90 goto 40
```

È ora il momento di approfondire un'altra caratteristica davvero notevole del VDC: la capacità di visualizzare più di 25 righe sullo schermo: tale caratteristica, unita alle 80 colonne, aggiunge ancora più flessibilità a un computer già fuori dal comune (per il 1985, s'intende).

Questa caratteristica era ben gradita a tutti i programmatori che volevano codificare elaboratori di testi o applicazioni con database e espandeva la matrice 80x25 (2000 caratteri sullo schermo) ad una matrice 80x28 (2240 caratteri); decisamente un enorme miglioramento rispetto allo standard 40x25 dell'epoca.





Cominciamo con un po' di teoria:

Nel registro 6 del controller video ("VERTICAL DISPLAYED"), si possono specificare quante linee mostrare a schermo. Il valore predefinito è 25. Cambiamo questo valore in 10:

```
10 A=DEC("D600"):D=A+1
20 POKE A,6:POKE D,10
```

Se eseguite il codice vedrete che il controller ora visualizza solo 10 linee sullo schermo e le linee rimanenti sono semplicemente "inghiottite". Proprio come possiamo rendere lo schermo più piccolo, abbiamo anche la possibilità di aumentare il numero di linee. Lo facciamo semplicemente correggendo la linea 20:

```
20 POKE A,6:POKE D,28
```

Ora possiamo (se il monitor è abbastanza buono) vedere tutte le 28 linee sullo schermo, anche se le ultime tre non contengono alcuna informazione utile.

Sappiamo già che la RAM video si trova all'indirizzo \$0000 e la RAM attributi all'indirizzo \$0800.

Dobbiamo cambiare questo dato perché ora dobbiamo visualizzare 2240 caratteri e quindi la fine della RAM video va spostata in avanti per evitare di sovrascrivere la RAM degli attributi (e viceversa). Nella memoria del VDC c'è abbastanza spazio tra la RAM degli attributi e il generatore di caratteri e possiamo spostare la seconda all'indirizzo \$0A00.

Ora però dobbiamo affrontare un altro problema perché non possiamo usare il BASIC per scrivere nelle righe aggiuntive.

L'interprete ottiene l'indirizzo di base della RAM degli attributi dall'indirizzo \$0A2F nella pagina zero; tecnicamente basterebbe solo informare l'interprete BASIC del nuovo indirizzo di base. Anche se logicamente corretto, non è così che funziona poiché se diamo un'occhiata più da vicino al kernal, vediamo che l'indirizzo di base non si ottiene tramite l'eventuale aggiunta di un offset ma tramite un'operazione logica "OR".

I bit 0 e 1 sono influenzati da ciò; questi due bit possono non essere rilevanti, cioè possono non essere impostati. Questo è il motivo per cui è consigliabile definire un nuovo indirizzo (\$1000) come indirizzo di partenza della RAM video. Facciamo questo con le due istruzioni seguenti, inserendo nel registro 20 ("ATTRIBUTE ADDRESS HI") il byte alto dell'indirizzo \$1000 tramutato in decimale (il decimale di \$10 è appunto 16):

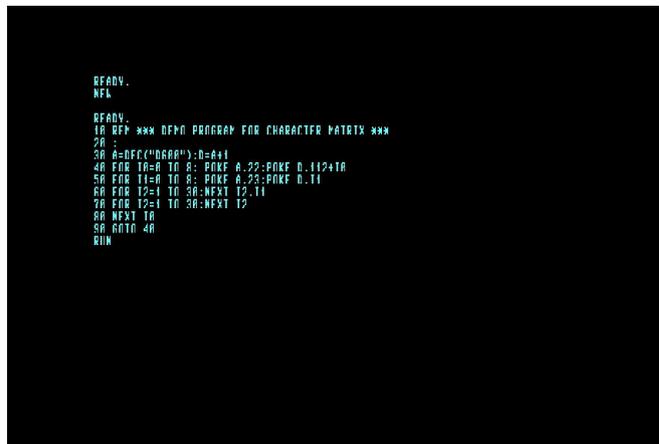


Fig. 4

```
POKE DEC("0A2F"),16
POKE DEC("D600"),20:POKE DEC("D601"),16
```

A questo punto tutto dovrebbe essere impostato: quando si avvia questo programma, sullo schermo appaiono 28 linee ma le ultime tre linee non hanno ancora un contenuto significativo. Sfortunatamente, come già accennato, non possiamo scrivere su queste righe con l'istruzione PRINT: il sistema operativo non è preparato per queste cose. Diventa chiaro che dobbiamo inserire dei caratteri (stringhe) in memoria. Questo viene fatto da una piccola routine di linguaggio macchina in modo che i caratteri da stampare possano essere prelevati da una stringa precedentemente memorizzata da qualche parte.

Analizzare il codice del linguaggio macchina per questa routine va oltre lo scopo di questo articolo, quindi, per il momento possiamo fare buon uso del seguente loader. Eseguite per caricare il codice LM in memoria e salvate il codice binario risultante per un uso successivo: la linea 120 vi aiuterà nel processo.

```
10 rem basic loader for print string
20 :
30 for i=dec("d00") to dec("d79")
40 read a$
50 poke i,dec(a$)
60 s=s+dec(a$)
70 next
80 if s<>16613 then print"error in data statements"
90 input"save program on diskette y/n";a$
100 if a$<>"y" then end
110 input"file name";f$
120 bsave""+f$+"",b1,p3328 to p3449:end
130 :
200 data 8e,00,d6,2c,00,d6,10,fb,8d,
01,d6,60,a2,12,a9,00
210 data
```



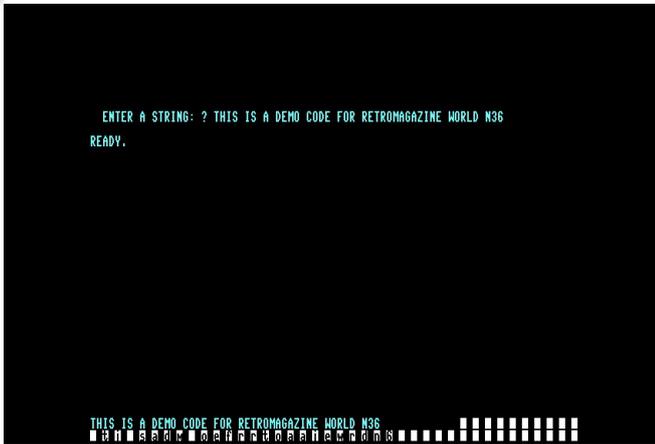


Fig. 5

```

20,00,0d,e8,a9,00,20,00,0d,a2,1f,a9,00,20,00,0d
220 data a2,12,a9,00,4c,00,0d,85,fc,
86,fd,a0,00,a2,01,a9
230 data fc,20,74,ff,85,fe,a0,01,a2,01,a9,fc,
20,74,ff,48
240 data c8,a2,01,a9,fc,20,74,ff,85,fd,
68,85,fc,a5,fc,d0
250 data
02,c6,fd,c6,fc,a5,fa,d0,02,c6,fb,c6,fa,a5,fa,85
260 data e0,a5,fb,85,e1,a2,01,a4,fe,a9,fc,
20,74,ff,a4,fe
270 data 84,ec,20,0c,c0,c6,fe,d0,e4,60

```

Fatto questo passo e con la routine LM in memoria, digitate il comando NEW per pulire l'area del BASIC e poi eseguite il programma seguente per accedere finalmente a uno schermo 80x28 hi-res completo, dove le ultime 3 linee in basso possono essere usate come linee di stato nei vostri programmi: le possibilità sono infinite! La figura 5 ne mostra una.

Fondamentalmente ciò che fa il codice è:

- ridefinisce i nuovi indirizzi di inizio delle sezioni della RAM del VDC (caratteri, attributi);
- definisce quante righe utilizzare (28, nel nostro caso);
- pulisce parte dello schermo stampando una serie di stringhe vuote per mezzo della routine LM;
- legge una stringa in input e la visualizza in una delle righe aggiuntive usando il comando BASIC "POINTER(nome variabile)" per recuperare l'indirizzo di dove la stringa è memorizzata nel banco 1.

```

10 rem *** demo program for 28 line screen ***
20 rem load and run previous program first
30 a=dec("d600"):d=dec("d601")
40 poke a,20:poke d,16:rem *** vdc gets new base
50 poke dec("0a2f"),16:rem *** kernal gets new
base address
60 poke a,7:poke d,28:rem *** 28 lines

```

```

70 poke a,6:poke d,33:rem *** new sync
80 :
90 print chr$(147);
100 t$="":rem 20 spaces
110 for x=0 to 79 step 20:for y=0 to 2
120 gosub 1000:next:next
130 input"enter your name: ";t$
140 for y=0 to 2:x=2*y:gosub 1000:next
150 end
1000 rem *** output t$ at x,y coordinate; y=0
means 1st line ***
1010 az=2000+y*80+x:rem destination address
1020 poke 250,az and 255:rem low byte
1030 poke 251,az/256:rem high byte
1040 t%=pointer(t$):rem address of the string
1050 sys dec("d27 "),t% and 255,t%/256: rem pass
1060 return
1070 :

```

Ancora una volta, questo breve articolo mostra quanto effettivamente avanzato fosse un computer che, al tempo, soffrì a causa della (ben meritata) reputazione del teoricamente sorpassato C64 nonché dell'ombra incombente dell'imminente introduzione di Amiga.

Per un vero 8-bit coder, però, programmare il C128 è ancora il Paradiso in Terra, poiché esso si dimostra ancora un'incredibile macchina "formativa" dove imparare e testare le proprie abilità. Restate sintonizzati!

#### Bibliografia:

Commodore 128 Internals, an authoritative insider's guide  
By K.Gerits, J.Schieb & F.Thrun  
A Data Becker Book Published by Abacus Software, 1985-1986





# Frattali in BASIC in salsa 8-bit - Parte 4

di David La Monaca

A dispetto di ciò che pensavo quando ho deciso di inaugurare questa piccola rubrica su RetroMagazine World, siamo giunti alla quarta puntata ed i nostri amati computer a 8-bit, pur con i loro limiti grafici, non smettono di stupirci. Altri contributi sono giunti in redazione da diversi lettori, sia lavori originali per nuovi frattali 2D, sia traduzioni di listati già pubblicati per linguaggi BASIC più o meno noti e per macchine più o meno famose. Insomma, il nostro obiettivo di suscitare nuovo interesse per la programmazione in BASIC a 8-bit, passando attraverso l'attrattiva di grafici in alta risoluzione e colori vivaci, è pienamente riuscito. E se continueranno ad arrivare proposte e listati interessanti dai lettori, potete star certi che raggiungeremo anche il quinto appuntamento di questa rubrica.

In questo numero il BASIC dello ZX Spectrum di casa Sinclair ottiene gran parte dello spazio, con listati che sono conversioni dei frattali apparsi nelle puntate precedenti o implementazioni ottimizzate per questa macchina, poiché ne sfruttano alcune locazioni di memoria particolari oppure sono scritte per cercare la massima velocità di tracciamento del grafico o per ottenere la maggiore compattezza del listato. Molti dei listati sono accompagnati da commenti e brevi descrizioni per facilitare la comprensione del codice e per modificarlo al fine di ottenere qualche effetto o qualche colore in più.

Ringraziamo quindi i vari autori di listati originali o delle conversioni da altri BASIC a 8-bit, tutti citati prima del codice, e senza ulteriore indugio partiamo con la rassegna dei frattali selezionati per questa rubrica.

## 20. ZX Spectrum BASIC – Spirale di Ulam

di Matthew Logue, dal gruppo FB "BASIC on ZX Spectrum"

Per maggiori info sulla spirale di Ulam consultare il numero precedente.

```
10 PLOT INVERSE 1;128,88: LET p=1: FOR a=1
TO 176: LET b=-a*(COS (a*PI)): LET x2=PEEK
23677: LET y2=PEEK 23678:
20 FOR x=(SGN b) TO b STEP SGN b: LET x1=x2+x:
LET p=p+1: FOR f=2 TO SQR p: IF p=f*INT (p/
f) THEN PLOT INVERSE 1;x1,y2: NEXT x: GO TO 35:
30 NEXT f: PLOT x1,y2: NEXT x
35 LET x2=PEEK 23677: LET y2=PEEK 23678:
FOR y=(SGN b) TO b STEP SGN b: LET p=p+1:
LET y1=y2+y: FOR f=2 TO SQR p: IF p=f*INT
(p/f) THEN PLOT INVERSE 1;x2,y1: NEXT y: NEXT a
```

```
40 NEXT f: PLOT x2,y1: NEXT y
50 NEXT a
```

## 21. ZX Spectrum BASIC – Spirale di Ulam e molto altro...

di Simon Férré, dal gruppo FB "BASIC on the ZX Spectrum"  
*Il listato può disegnare la spirale di Ulam ed altre spirali plottando solo un punto sullo schermo ogni N calcolati. Cambiando il valore della variabile c nella linea 10 verrà modificato il "passo" per disegnare una spirale normale. Cambiando il valore di pr nella linea 10, il programma disegnerà la Spirale di Ulam (pr=1) oppure una normale (pr=0). Un valore di 2 per la variabile c produrrà una trama a scacchiera, mentre un valore pari a 8 genera due trame diverse da entrambe le parti di una diagonale immaginaria che parte dall'origine (il centro dello schermo).*

```
10 LET c=7: LET x=127: LET y=88: LET md=1:
LET n=1: LET p=1: LET pr=1
20 FOR h=1 TO ABS md
30 IF pr=0 AND p=c THEN IF x>=0 AND x<=256
AND y>=0 AND y<=175 THEN PLOT x,y
40 IF pr=1 THEN IF x>=0 AND x<=256 AND y>=0
AND y<=175 THEN GO SUB 140: IF NOT d THEN
PLOT x,y
50 LET n=n+1: LET p=p+1-(p AND p=c): LET
x=x+(1 AND md>0)-(1 AND md<0): NEXT h
60 IF n>=65536 THEN GO TO 130
70 FOR v=1 TO ABS md
80 IF pr=0 AND p=c THEN IF x>=0 AND x<=256
AND y>=0 AND y<=175 THEN PLOT x,y
90 IF pr=1 THEN IF x>=0 AND x<=256 AND y>=0
AND y<=175 THEN GO SUB 140: IF NOT d THEN
PLOT x,y
100 LET n=n+1: LET p=p+1-(p AND p=c): LET
y=y+(1 AND md>0)-(1 AND md<0): NEXT v: LET
md=-1*(md+(1 AND md>0)-(1 AND md<0))
110 IF n>=65536 THEN GO TO 130
120 GO TO 20
130 STOP
140 LET d=0: IF n=1 THEN LET d=1: RETURN
145 IF n=2 OR n=3 THEN RETURN
150 IF n/2=INT (n/2) THEN LET d=1: RETURN
155 IF n<17 THEN FOR i=3 TO n-1 STEP 2: LET
d=d+(n/i=INT (n/i)): GO TO 165
160 FOR i=3 TO SQR (n) STEP 2: LET d=d+(n/
i=INT (n/i))
165 IF d=0 THEN NEXT i
```





```
170 RETURN
```

## 22. ZX Spectrum/ZX81 Basic – Spirale di Pierluigi Pieroli, dal gruppo FB Z80

```
10 INPUT N: PRINT AT 1,6;N
20 LET D=2*PI/600
30 LET P=84
35 PLOT 128,88
40 FOR I=0 TO 600
50 LET T=D*I
55 LET R=P*SIN (N*T)
60 LET X=R*COS T+128
70 LET Y=R*SIN T+88
80 DRAW X-PEEK 23677,Y-PEEK 23678
90 NEXT I
```

Sostituire/aggiungere le seguenti linee per ottenere un po' di colore:

```
5 PAPER 0: BORDER 0: CLS
80 INK x/32+1: DRAW x-PEEK 23677, y-PEEK 23678
```

### 23.a Quick Basic 3.5 - PC IBM/DOS - Spirale di Archimede

```
10 REM ***ARCHIMEDES SPIRAL***
30 SCREEN 11 : CLS : PI=3.141592654
40 WINDOW (-12,-9)-(12,9)
50 A=.1 : PSET(0,0)
60 FOR T=0 TO 16*PI STEP .1 : R=A*T
70 X=R*COS(T) : Y=R*SIN(T)
80 LINE -(X,Y)
90 NEXT T
100 A$=INPUT$(1) : END
```

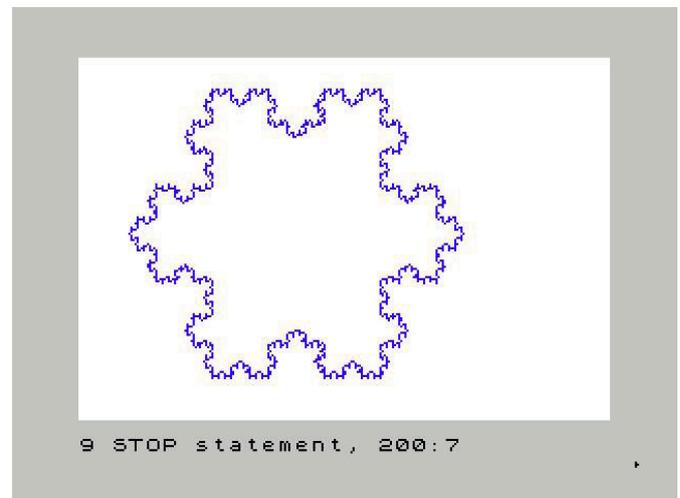
### 23.b Quick Basic 3.5 - PC IBM/DOS - Spirale logaritmica

```
10 SCREEN 11: CLS : PI = 3.141593
20 WINDOW (-4, -3)-(4, 3)
30 A = .05: B = .1 'Start point & growth rate
40 PSET (A, 0)
50 FOR T = 0 TO 42 STEP .1: R = A * EXP(B * T)
60 X = R * COS(T): Y = R * SIN(T)
70 LINE -(X, Y)
80 NEXT T: A$ = INPUT$(1): END
```

## 24. ZX BASIC – Triangolo di Sierpinsky

```
10 PAPER 0: BORDER 0: BRIGHT 1: CLS
20 LET x=160*RND: LET y=140*RND
25 LET ox=47: LET oy=157: REM xy offsets
30 LET c=INT(3*RND)
```

```
40 LET x=x/2: LET y=y/2
50 IF c=0 THEN LET x=x+40: LET y=y+70: GO TO 70
60 IF c=1 THEN LET x=x+80
70 PLOT INK 6-c;ox+x,oy-y
80 GO TO 30
```



## 25. ZX SPECTRUM BASIC (Z80, 1984) – Fiocco di neve di Koch

```
10 LET l$="F--F--F"
20 FOR i=1 TO 3
30 LET n$=""
40 FOR j=1 TO LEN l$
50 IF l$(j)="F" THEN LET n$=n$+"F+F--F+F":
GO TO 70
60 LET n$=n$+l$(j)
70 NEXT j
80 LET l$=n$
90 NEXT i
100 LET ang=0
110 PLOT 174,5
120 FOR i=1 TO LEN l$
130 IF l$(i)="F" THEN DRAW 6*SIN ang,6*COS ang
140 IF l$(i)="+" THEN LET ang=ang+PI/3
150 IF l$(i)="-" THEN LET ang=ang-PI/3
160 NEXT i
```



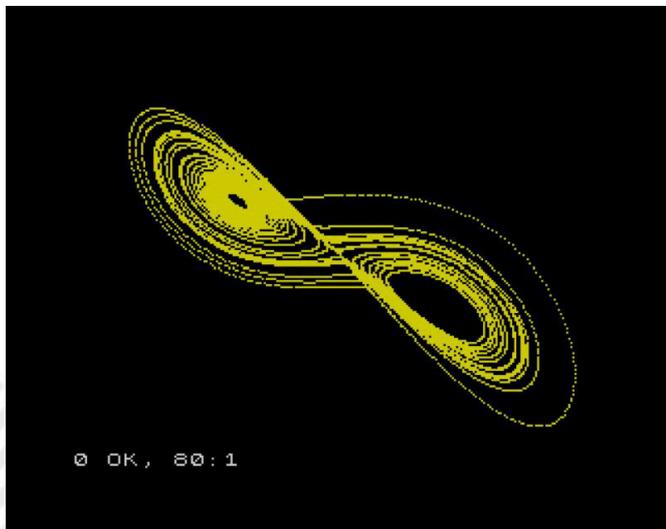
**26. VIC-20 + SUPER EXPANDER JP (6502, 1981) -****Gomitolo / Woolball**

Conversione e adattamento: Armando Pavese, Biella (Italia)

```

10 hires0
15 setc1,4,6
110 dimx(16):dimy(16)
120 fori=1to4
130 forj=1to4
140 k=4*i+j-4
150 x(k)=j-3:y(k)=i-3
160 nextj,i
170 x(2)=0:y(2)=-3
180 x(8)=2:y(8)=0
190 x(9)=-3:y(9)=-1
200 x(15)=-1:y(15)=2
210 fori=1to16
220 forj=1to16
230 fork=1to16
240 xx=16*x(i)+4*x(j)+x(k)
250 yy=16*y(i)+4*y(j)+y(k)
260 plotxx+85,-yy+75
270 nextk,j,i
280 setc3:char1,1,"vic20 gomitolo"
300 wait198,1:poke198,0
310 text:setc7,2,4:end

```

**27. ZX Basic - Lorenz Attractor**Conversione da Dragon BASIC di **Anja de Weerd** (Eindhoven, Paesi Bassi)

```

10 REM PROGRAM Lorenz system *
20 BORDER 0: PAPER 0: INK 6: CLS
30 LET rho=10: LET sigma=28: LET beta=8/3:
REM Constants for this type of Lorenz system
35 LET d=.002: REM Plotting density/distance
40 LET x=.1: LET y=0: LET z=0: REM Starting

```

values for the coordinates

```

50 FOR i=1 TO 16000
60 PLOT 127+5.7*x,87-3*y
70 LET x=x+d*rho*(y-x): LET y=y+d*(x*(sigma-
z)-y): LET z=z+d*(x*y-beta*z): REM Algorithm
to change the coordinates
80 NEXT i

```

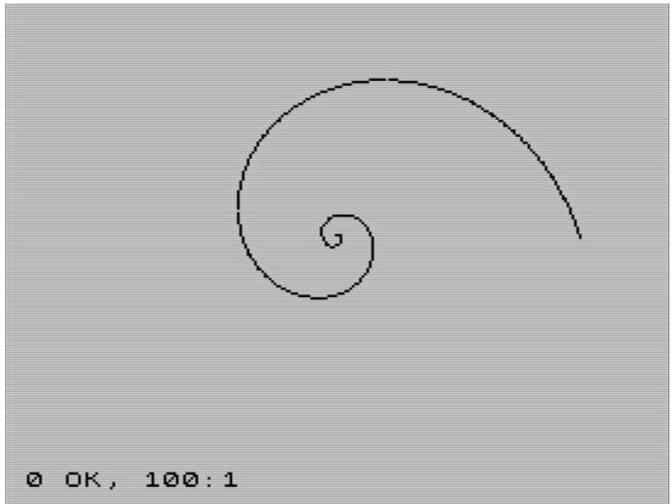
**28. ZX Basic – Spirale di Archimede**di **Uwe Geiken**, Germania

*Il fattore a determina quanto siano vicine le linee. La variabile r è il raggio che cresce linearmente, p l'angolo ed s è lo step per l'angolo. Dopo un cerchio complete, si calcola un nuovo passo in modo che i punti siano vicini (linea 35).*

```

10 LET a=.6: LET r=1: LET p=0: LET s=.2
20 LET x=128+r*COS p: IF x>=0 THEN IF x<=255
THEN LET y=88+r*SIN p: IF y>=0 THEN IF y<=175
THEN PLOT x,y
30 LET p=p+s: LET r=r+a*s
35 IF p>2*PI THEN LET p=p-2*PI: LET s=.8/r
40 GO TO 20

```

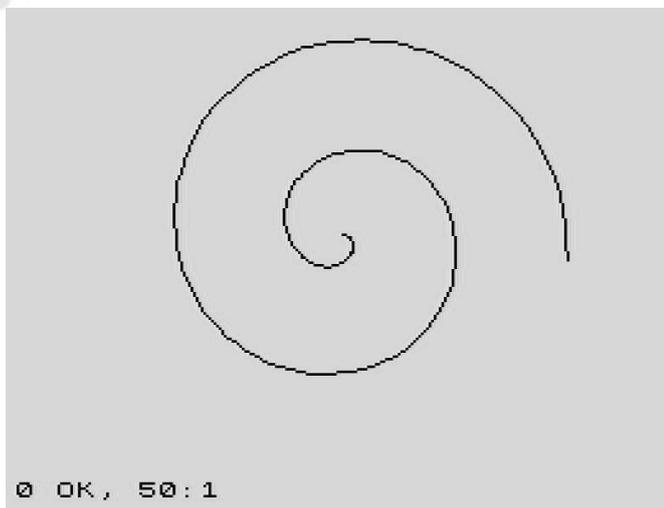
**29. ZX Basic – Spirale di Fibonacci**di **Richard Scaglione**, dal gruppo FB "BASIC on ZX Spectrum"

```

10 LET p=2*COS (PI/5)
20 LET c=0
30 LET r=100
40 LET x=127+r*COS c
50 LET y=87+r*SIN c
60 PLOT x,y
70 LET d=ASN (1/r)
80 LET r=r*((1/p)^(d/(PI/2)))
90 LET c=c+d
100 IF INT r THEN GOTO 40

```





### 30. ZX Basic – Spirale di Fibonacci

di **Slavo Labsky** – ottimizzato per la velocità

*Utilizza le locazioni 23677 e 23678 dello Spectrum per recuperare l'ultimo punto tracciato.*

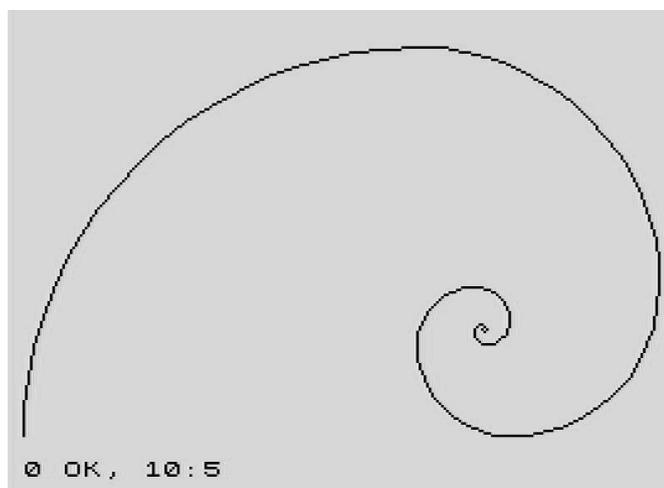
```
10 PLOT 216,78
20 FOR r=88 TO 0 STEP -.5
30 LET u=r*.144
40 DRAW 128+r*COS u-PEEK 23677,88-r*SIN u-
PEEK 23678
50 NEXT r
```

### 31. ZX Basic – Spirale aurea

di **Uwe Geiken**, Germania

*Se modificate lo STEP al valore 1, vedrete che i punti avranno tutti la stessa distanza fra loro.*

```
54 FOR r=1 TO 189 STEP .2: LET p=LN (r*4.175)/.
3: PLOT 183+r*COS p,45+r*SIN p: NEXT r
```



### 32. ZX Basic – Spirale aurea 1

di **Matthew Logue**, UK – versione con DRAW (120 bytes)

```
10 LET a=255: LET b=1.618: FOR x=1 TO 12:
DRAW (((INT ((x+1)/2)-2*INT ((x+1)/4))*2)-
1)*(a/b^x), (-((INT ((x)/2)-2*INT ((x)/
4))*2)-1)*(a/b^x)), -PI/2: NEXT x
```

### 33. ZX Basic – Spirale aurea 2

di **Matthew Logue**, UK – versione con PLOT (57 bytes)

```
10 FOR a=1 TO 19.1 STEP .001: LET r=-.615*EXP
(.3*a): PLOT r*COS a+183,r*SIN a+45: NEXT a
```

Con questo articolo, la rubrica, iniziata un po' per gioco e un po' per stimolare i nostri lettori a rimettere le mani sulla tastiera delle loro macchine a 8-bit, ha raggiunto il suo termine (ed anche il suo obiettivo). In futuro, potremmo riprenderne la pubblicazione se riceveremo altri input e contributi dai lettori e dagli appassionati di grafica generata a partire da formule matematiche. D'altronde si tratta pur sempre di un territorio davvero enorme da esplorare.

Come al solito, vi invitiamo a mandare idee, spunti di riflessione e listati, magari corredati da qualche riga di informazioni e/o istruzioni alla e-mail: [RetroMagazine.Redazione@gmail.com](mailto:RetroMagazine.Redazione@gmail.com).

Tutti coloro che scriveranno otterranno una risposta e, se raggiungeremo una massa critica interessante, organizzeremo una nuova puntata di "Frattali in BASIC in salsa 8-bit" nei prossimi numeri della rivista. Per il momento grazie ancora a tutti coloro che hanno contribuito o partecipato alla buona riuscita di questa serie e un arrivederci a presto con altri intriganti argomenti relativi alla grafica 8-bit!





# Intervista ad Andrea Contato

di Alessandro Albano

Un po' per caso ho conosciuto il protagonista di questa intervista, dopo aver risposto ad un suo post su Facebook. Parlava di un suo libro sui videogames, diverso dal solito. Incuriosito, l'ho contattato ed ho così cominciato a scrivere con lui su Messenger. Mi ha colpito molto la sua passione e disponibilità. Così ho cominciato a leggere il suo libro, anzi, ho cominciato ad ascoltarlo dato che sono non vedente. Ora vi svelo il suo nome... Andrea Contato è stato da subito sensibile alla tematica dell'accessibilità e devo dire che il suo libro in formato epub è stato da subito "senza barriere", facendo appassionare anche me, più colpito dai contenuti rispetto che dalle tante immagini. Così mi sono detto che questa produzione sarebbe potuta piacere davvero a tanti ed ho deciso di intervistarlo. A me, "Storie di videogiochi..." ha appassionato, l'ho letto tutto d'un fiato e non vedo l'ora di leggere il seguito. Ed ora lasciamo la parola ad Andrea.

## Ciao Andrea, ti va di parlarci un po' di te?

Sono un videogiocatore. Lo sono sempre stato e presumibilmente lo sarò anche nel futuro. Oggi ho meno tempo per giocare, anche perché mi dedico molto alla ricerca e alla scrittura, ma quando i giochi di cui spesso parlo erano appena usciti ed erano lo stato dell'arte, ero giovane e avevo molto tempo libero per provarli tutti.

Sono nato nel 1976, quindi ho perso la primissima generazione di videogiochi. L'ho vista solo in differita, sotto forma di vecchi cabinati semi dismessi nei bar di paese. Ma la prima grande ondata di console e giochi per computer, almeno qui in Italia, è arrivata quando ero abbastanza grande da saper leggere e scrivere e godermela tutta.

Ho sempre avuto tre grandi passioni: i videogiochi, la storia e la lettura/scrittura.

Ad un certo punto ho trovato la quadra e ho combinato questi tre interessi in una sola attività: fare il ricercatore, lo storico dei videogiochi e scrivere libri sulla storia dei videogiochi.

## Parlacene.

Il mio primo esperimento è stato *Through the Moongate*: la storia di Richard Garriott, Origin Systems Inc e Ultima.

L'impresa nacque dalla constatazione che non esisteva nessuna monografia su Garriott, la sua compagnia più importante e la serie di giochi di ruolo per computer (una delle più longeve ed importanti della storia). Sì, c'era una vecchia opera scritta da Shay Addams, ma era un lavoro commissionato direttamente da Origin e Garriott stesso, interessante, ben scritto, ma certamente non obiettivo. Ho deciso di colmare io questa lacuna e mentre mi mettevo all'opera, pian piano è emerso un progetto più grande: non accontentarsi di rimaneggiare e ordinare le informazioni già disponibili, ma andare alla fonte e cercare nuovi punti di vista. Così ho iniziato un lungo giro di interviste e più parlavo con sviluppatori che in genere non venivano mai considerati da giornalisti e scrittori, più mi rendevo conto che c'era una parte di storia che non era mai stata raccontata.

## Cioè?

*Through the Moongate* è stato il punto di partenza di questa mia riflessione a cui poi è seguito *Video-Giochi*, il mio nuovo lavoro. Si tratta di un'opera in cinque volumi

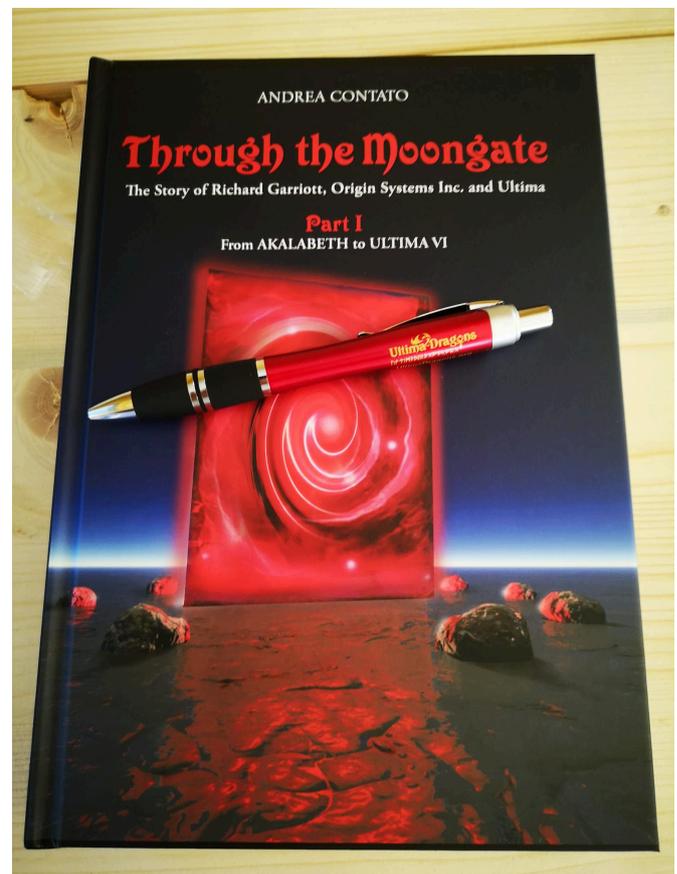
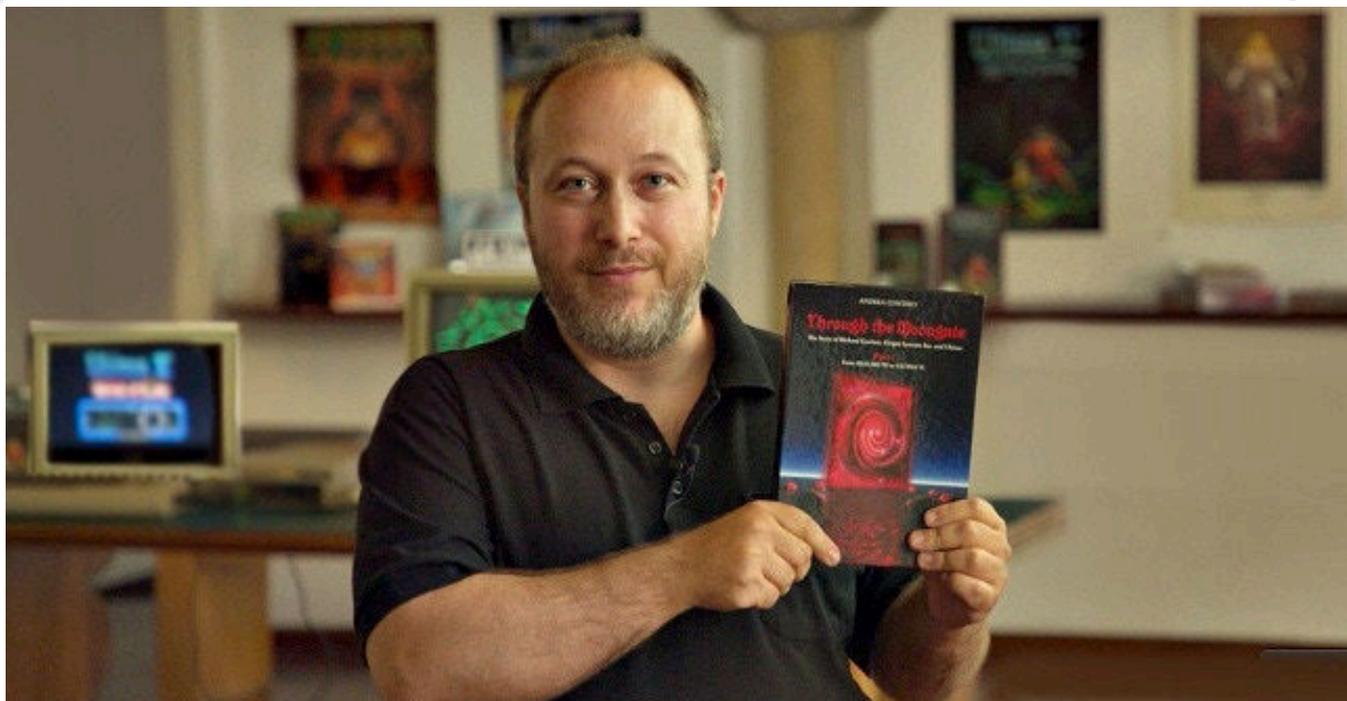


Fig. 1 - *Through the Moongate*





**Fig. 2 - Andrea Contato con il suo libro 'Through the Moongate'**

che ripercorre la storia dei videogiochi dalle origini fino al 1999, nella sua interezza (giochi per mainframe, minicomputer, microcomputer, console, arcade, handheld), con un'enfasi particolare sull'aspetto umano e creativo. In Video-Giochi metto al centro dell'attenzione, come avevo iniziato a fare in Through the Moongate, gli sviluppatori con le loro esperienze, le loro capacità, i loro gusti personali, il rapporto con la tecnologia e le scelte fatte nel creare giochi. Spesso quando si parla di storia dei videogiochi si pone l'attenzione sulla questione economica, sulle compagnie in competizione, sulle console vendute e le cartucce commercializzate. Video-Giochi è, prima di tutto, una racconta di storie umane: persone alle prese con la creazione di giochi; persone che, spesso, prima di sviluppare giochi, hanno giocato quelli di altri, si sono divertiti, si sono appassionati e, alla fine, hanno deciso di fare il loro gioco, spesso traendo ispirazione dal lavoro degli altri.

#### **Un'opera ambiziosa.**

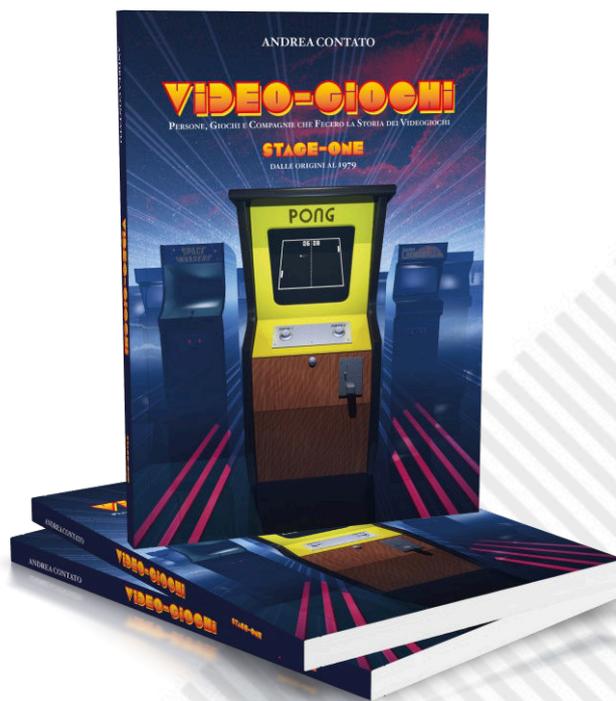
Sì, decisamente. Infatti ho rinunciato a comprimere tutto in un solo volume e ho deciso di partire già con l'idea di produrne cinque. Sperando bastino. Perché va bene fare sintesi, ma ci sono tante cose da raccontare e non parlo di aneddoti, ma di informazioni che, se pure divertenti e curiose, servono per dipingere un'epoca che è stata e che non tornerà, un modo di fare giochi che ormai non esiste più e che ci dicono molto su come mai quelli che oggi consideriamo i grandi classici, sono stati fatti così e non in altro modo, come mai hanno avuto successo e

si sono distinti dal resto del mercato.

Ci vuole tempo e spazio per raccontare tutto e, grazie al fatto che questo è un prodotto indipendente, non ho dovuto scendere a patti con nessuno.

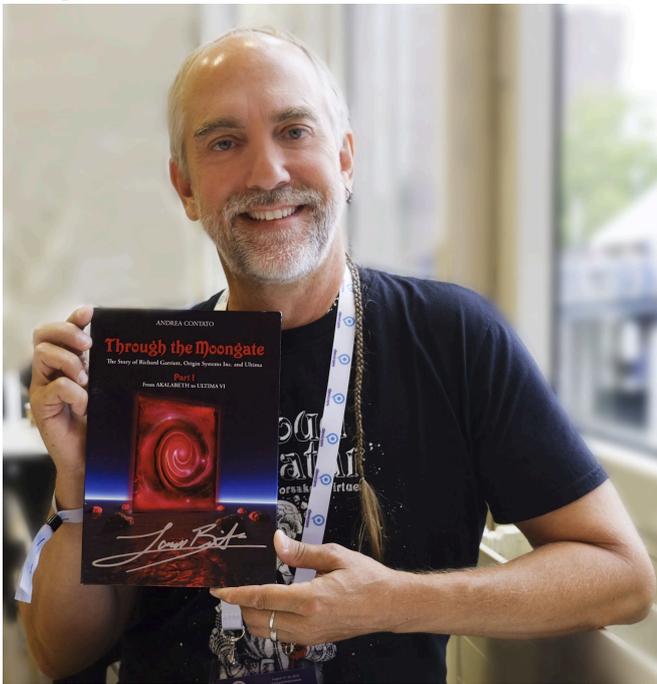
#### **Dove possiamo reperire questo libro?**

Il primo è già disponibile e copre l'arco temporale dalle origini dei videogiochi fino al 1979. I volumi successivi, che sono in corso di stesura e che dovrebbero uscire con cadenza annuale, già a partire dal 2022, trattano cinque anni alla volta, dal 1980 fino a 1999.



**Fig. 3 - Video-Giochi stage-one**





**Fig. 4 - Richard con il libro**

Chi volesse acquistarne una copia, può ordinarla direttamente dal mio store ufficiale all'indirizzo [www.andreacontato.com/negoziario](http://www.andreacontato.com/negoziario), dove può trovare anche gli altri libri, o dal sito ufficiale dell'opera Video-Giochi [www.retro-game.it](http://www.retro-game.it) o su Amazon. In quest'ultimo caso, però, vale la pena di far notare che la versione ebook è solo in formato MOBI e che l'edizione cartacea Print on Demand è leggermente più piccola rispetto a quella stampata da noi.

#### **Noi?**

Sì, Video-Giochi è edito da Retroedicola Videoludica, un'associazione culturale con sede a Bergamo e che ha come scopo la divulgazione e la preservazione della cultura e della storia del videogioco.

#### **Obiettivi simili ai tuoi.**

Direi proprio di sì, infatti è nata subito una collaborazione fruttuosa. Retroedicola da anni scansiona i giornali di videogiochi e informatica. Ultimamente, grazie anche al notevole talento dei suoi membri, ha intrapreso la vita della ricostruzione di alcuni dei primissimi videogiochi tra cui Tennis For Two, OXO e Nimrod. C'è un sito ufficiale per queste ricostruzioni, [www.rebuildingbits.it](http://www.rebuildingbits.it)

#### **Come e quando è iniziata la tua passione per i videogiochi?**

Come? Per caso. Quando? Nei primi anni '80. Con la mia famiglia eravamo in viaggio per l'Europa con un furgoncino Volkswagen Type 2, un vero cliché. Facemmo tappa in Olanda e mi intrufolai in un bar del camping. C'era un

cabinato di Asteroids che attirò subito la mia attenzione. Mi parcheggiai lì per ore, forse giorni, a guardare le partite degli altri. Quando provai a giocare io mi schiantai subito e tornai a fare da spettatore entusiasta e pronto a dare suggerimenti, in italiano, ai giocatori internazionali. Devo essere stato un gran rompiscatole.

Poi tornammo a casa e non ebbi più esperienze con i videogiochi per qualche tempo. Gli arcade erano cose da bar e i bar, nei primi anni '80 non erano locali per bambini in età prescolastica. Ma nel 1983 circa entrò in casa mia un gioiellino tutto italiano, una piccola console domestica. Si chiamava Universal Game Computer e la producevano a Curno, in provincia di Bergamo. A pochi chilometri da casa mia! Ma allora non lo sapevo. L'ho scoperto molti anni più tardi, cosa che, tra l'altro, mi permette di anticipare una delle mie prossime imprese.

#### **Ovvero?**

Recuperare la storia della Cabel di Curno, una delle più importanti aziende italiane operanti nel settore dei videogiochi. Piccola fabbrica di elettrodomestici e televisori che per quasi cinque anni produsse anche console. E ne vendette molte che ora sono oggetti da collezionismo.

#### **Quale è la sfida più difficile per chi ricerca la storia dei videogiochi?**

Quando ho iniziato qualche anno fa, ti avrei risposto: trovare le fonti, reperire le persone, riuscire a contattarle, intervistarle, farsi dare le risposte di cui hai bisogno per colmare i buchi. In una decina di anni è cambiato tutto. Qualche tempo fa il retrogaming e la storia dei videogiochi erano argomenti considerati di scarso interesse e molti sviluppatori erano diffidenti quando venivano contattati. Non capivano come mai ci fosse tanto interesse per la loro vita e le loro opere.

Oggi è cambiato tutto. C'è un interesse crescente e gli sviluppatori lo sanno. Quando li contatto sono spesso molto disponibili, anche perché tanti di loro sono ormai in pensione e anziani. Hanno quindi il tempo che serve per frugare nella memoria, ricordare e raccontare la loro storia.

#### **Quindi quale è la risposta oggi?**

Il problema è fare in fretta. I primi pionieri dell'informatica e dei videogiochi sono già molto anziani o purtroppo ormai deceduti. Bisogna far presto per conservare la memoria prima che svanisca per sempre. C'è tanto da indagare, poco tempo a disposizione. Nel nostro piccolo, io e





Fig. 5 - Il segnalibro di Through the Moongate

Retroedicola Videoludica stiamo facendo il possibile per conservare tutto quello che riusciamo.

### Progetti per il futuro?

Finire Video-Giochi. Ho un volume da completare e tre da scrivere da zero. Il piano dell'opera esiste già, anche se è provvisorio e sicuramente sarà modificato, e prevede di trattare di tutti i giochi, le compagnie e l'hardware più importanti fino al 1999. Stiamo inoltre lavorando alla traduzione inglese dell'opera. Con Through the Moongate abbiamo visto che all'estero c'è molto interesse per queste opere e il fatto che siano scritte da un autore italiano non è un grande ostacolo. La lingua, però, sì e c'è molto lavoro da fare.

Entro l'estate, inoltre, insieme ad Enrico Ricciardi, che già mi ha aiutato a pubblicare Through the Moongate, sto scrivendo un libro sulla storia dei primi anni dei giochi per microcomputer, dal 1977 al 1980 circa, e per farlo siamo andati a scovare i pionieri che in quegli anni hanno gettato le fondamenta per l'industria dei giochi per computer: Scott Adams, i coniugi Williams, Richard Garriott, Robert Clardy e molti altri ancora. A nostro avviso, si tratterà di un'opera speciale perché creata con due obiettivi diversi e che ben si combinano: oltre a contenere le mie ricerche storiche e le mie interviste focalizzate sulla creazione dei giochi e sui problemi tecnologici affrontati, avrà un incredibile approfondimento curato da Enrico, nel quale si affronterà con precisione e dettaglio l'aspetto

collezionistico e produttivo di questi giochi – che ormai sono divenuti degli ambiti e preziosi cimeli – con una sezione fotografica senza precedenti.

Faremo un Kickstarter, tra marzo e maggio.

Se l'argomento è di interesse, potremmo parlarne nuovamente in futuro

Grazie per questa interessante intervista. Sono certo che sentiremo ancora parlare molto di Andrea in futuro, se avete alcune curiosità non esitate a contattarlo.

### Links utili:

**Sito ufficiale:** [www.andreacontato.com](http://www.andreacontato.com)

**FB ufficiale:** <https://www.facebook.com/Andrea-Contato-343821635963825>

**Twitter:** <https://twitter.com/xender76>

**Sito ufficiale di Through the Moongate:** [www.theira.it](http://www.theira.it)

**FB ufficiale di TTM:** <https://www.facebook.com/Through-the-Moongate-216842488907011>

**Sito ufficiale di Video-Giochi:** [www.retro-game.it](http://www.retro-game.it)

**FB ufficiale di Video-Giochi:** <https://www.facebook.com/RetroedicolaEditore>

**Youtube di retroedicola con dirette sulla storia dei**

**videogiochi:** <https://www.youtube.com/playlist?list=PL2Iy4uB19ZctInWL1YbEW-XvGRIBPYGrI>

**Kickstarter di LOAD AND PLAY:** [https://](https://www.kickstarter.com/projects/1108065491/the-inside-story-of-the-birth-of-microcomputer-games)

[www.kickstarter.com/projects/1108065491/the-inside-story-of-the-birth-of-microcomputer-games](https://www.kickstarter.com/projects/1108065491/the-inside-story-of-the-birth-of-microcomputer-games)





# Diamo un'occhiata a Lykia – The Lost Island

di Carlo Nithaiah Del Mar Pirazzini

Questo progetto nasce dagli autori di Pets Rescue e Alpharay, due ottimi titoli.

Lykia è un action rpg che ricorda tantissimo i classici per Super Nintendo come Secret of Mana e The Legend of Zelda: a Link to the Past.

Il gioco è in arrivo per Commodore 64, Plus/4 e C16 espanso e sarà disponibile in versione digitale e fisica.

La storia ci racconta di Nora e del suo 16° compleanno, una festa che si trasformerà in qualcosa di epico.

Il tutto parte dall'invito da parte di sua madre a raccogliere frutti di Apothia, una pianta molto particolare. Mentre Nora raccoglie il frutto accade qualcosa che colpisce tutto il villaggio e la natura circostante.

Ha anche dotato Nora di poteri magici.

Insomma, un'introduzione classica per questo tipo di giochi. Abbiamo raggiunto via social Stefan Mader, coder e mente dietro al progetto e ci abbiamo fatto un paio di chiacchiere. Stefan è una persona molto alla mano e cortese e ci ha

inviato una demo completa per dare un'occhiata al titolo, ma ne parleremo dopo.

Lascio il campo libero a Stefan che ci racconterà del suo titolo.

*“Lo sviluppo è nato nel 2020 circa... Abbiamo pensato col team ad un genere di gioco che difficilmente si vede su Plus/4 e su macchine a 8 bit.. e poi io adoro Zelda! Le cose messe assieme hanno dato vita ad un gioco test per vedere se effettivamente si poteva procedere col progetto ed è nato Lykia.*

*Nel team siamo io dalla Germania, Robert Kisnemeth dall'Ungheria che si è occupato dei tileset grafici, Ronny Doll per le musiche su Plus/4 sempre dalla Germania, così come Markus Jentsch per la musica su Commodore 64. Poi ci sono alcuni sviluppatori che si occupano di eseguire le routine del framework come Ingo Jache, Imre Szell e*





*Balazs Szabo. Si occupano di rendere il titolo il più compatibile possibile con flashcard, SD2IEC e altro.*

*In realtà il team lavora insieme già da diverso tempo. Siamo collaudati!*

*A proposito di date d'uscita... in realtà abbiamo quasi finito. Direi che siamo circa al 99% del prodotto finale. Tuttavia, l'ultimo 1% sembra occuparci davvero molto tempo. Quindi purtroppo non riesco a darvi una data precisa per l'uscita del gioco. Facciamo del nostro meglio ma è un progetto nel nostro tempo libero e molti di noi hanno una famiglia, il lavoro e altre cose da mettere davanti... e non posso di certo costringere nessuno a fare le cose prima. Però ci siamo. Abbiate fiducia!"*

Le parole di Stefan sono chiare e attendiamo con fiducia l'arrivo della versione finita.

Parlando del gioco vero e proprio che abbiamo visionato c'è da dire che il lavoro fatto fino ad ora è notevole.

Una bellissima introduzione racconta la storia del gioco e ricorda proprio i titoli per le console Nintendo.



Bella anche la scelta grafica sia su Commodore 64 che su Plus/4 (quest'ultima mi è apparsa persino più "vivida" nei colori e nelle animazioni).

La colonna sonora è molto "europea" e si sente. È uno stile perfettamente integrato nel tipo del gioco ma dalla forte connotazione geografica.

Interessante anche il sistema di menù richiamabile tramite tastiera. Ricorda quello di Zelda ed è abbastanza semplice. Si seleziona l'oggetto e si clicca il tasto di fuoco per equipaggiarlo. Semplice.

La zona da esplorare è molto vasta come sono molto caratteristici i nemici e i dungeon. Bello.

Abbiamo testato il gioco su emulazione (VICE) e non abbiamo riscontrato nessun problema.

Così come per quanto riguarda l'utilizzo su un vero Commodore 64 con SD2IEC e sul THEC64.

Insomma, non ci resta che aspettare il gioco finale che verrà pubblicato da Psytronik.

Potete seguire lo sviluppo anche da qui:

<http://www.psytronik.net/newsite/index.php/c64/140-lykia64>





# War and Videogames

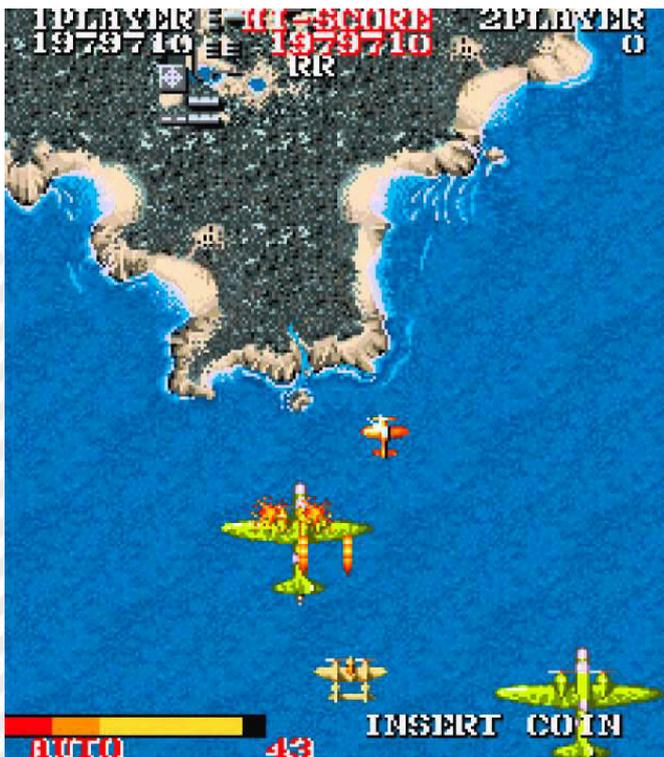
di Mic the Biker Novarina

L'inizio della Seconda guerra mondiale viene indicato da gran parte della storiografia il 1° settembre del 1939. Mi piace dirlo e ripeterlo sempre: "Historia magistra vitae est". È nostro dovere ricordare queste date, ricordare le nefandezze che una guerra (qualunque essa sia) si porta appresso. E anche noi nel nostro piccolo, vogliamo farlo, ma ovviamente in maniera differente. Tutti ricordano i giochi ambientati nel periodo della guerra usciti negli ultimi 15 anni, sia per PC che per le console che si sono succedute. Ma i giochi a sfondo bellico hanno radici antiche che magari qualcuno può aver dimenticato.

Molti di essi hanno preso spunto da eventi svoltisi durante la guerra e i primi due che voglio ricordare sono gli immortali 1942 e 1943: The Battle of Midway di casa Capcom. La battaglia delle Midway fu un vero flop per la flotta giapponese, la quale era salpata per un'imponente operazione nel Pacifico centrale avente come obiettivo l'occupazione dell'atollo di Midway. Gli statunitensi erano tuttavia perfettamente al corrente della mossa nemica grazie alla decifrazione dei codici crittografici giapponesi portata avanti dal sistema "Magic". La battaglia delle Midway tra il 4 e il 6 giugno 1942 rappresentò il punto di

svolta della guerra nel Pacifico. Le quattro portaerei dell'ammiraglio Nagumo furono sorprese dai bombardieri statunitensi e colate a picco nel giro di pochi minuti. Il primo gioco della serie è **1942**, uscito in sala giochi nel 1984 e poi convertito su innumerevoli piattaforme. Si tratta di uno sparattutto verticale, come andava molto di moda in quel periodo. Il protagonista comanda un Lockheed P-38 Lightning con lo scopo di raggiungere e bombardare Tokyo. Durante il lungo percorso sorvolerà varie località dell'Oceano Pacifico: Midway, Marshall, Attu, Rabaul, Leyte, Saipan, Iwo Jima e da ultimo Okinawa. Ovviamente il percorso non sarà privo di pericoli, visto il gran numero di mezzi giapponesi che fanno fuoco da ogni parte. Nel 1987 vede la luce il sequel, chiamato **1943: The Battle of Midway**. Questo episodio resta, a mio avviso, il migliore di tutta la serie. Concentrava la sua trama sulla battaglia di Midway. La novità più significativa fu la possibilità di giocare in due contemporaneamente, cosa che rese questo cab una vera superstar di quegli anni. Era veramente una cosa pazzesca mettersi spalla a spalla con l'amico fidato e insieme imbastire tattiche blastanti mentre si esibiva il celeberrimo loop!

Nel frattempo, anche sul fronte home computer il tema era più caldo che mai. Un'autentica perla era **Aces of Aces**, pubblicato nel 1986 per tutte le piattaforme 8 bit da tre diverse società: Accolade, US Gold e Tiertex Design Studios. Ace of Aces è un simulatore di volo da combattimento ambientato nella Seconda guerra mondiale. Il giocatore vola a lungo raggio su un caccia-bombardiere equipaggiato

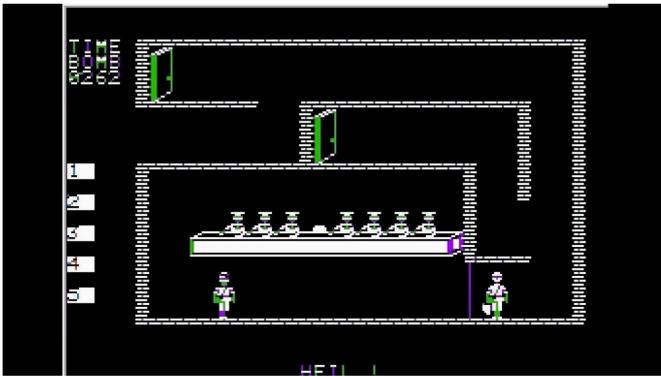


1943: The Battle of Midway



Aces of Aces





**Beyond Castle Wolfenstein**

di tutto punto. A bordo abbiamo missili, bombe e un cannone mitragliatore. Le missioni includono la distruzione di aerei da combattimento tedeschi, bombardieri, bombe volanti V-1, U-boat e treni. Esistono cinque diverse opzioni di visualizzazione: la cabina di pilotaggio, le ali destra e sinistra, la mappa di navigazione e il vano bombe. Tutto selezionabile utilizzando la tastiera o toccando due volte il pulsante di fuoco e spostando il joystick nella direzione desiderata. Fu uno dei giochi più venduti per Commodore 64, il secondo in assoluto di casa Accolade, e ancora oggi si fa giocare molto volentieri. Questo perché la componente simulativa era sì presente ma, durante le missioni, si partiva già in volo, eliminando lo stress di dover far alzare un bombardiere da terra ma soprattutto farlo atterrare senza danni.

Sto sentendo molti di voi brontolare: scommetto che a quasi tutti, parlando di Seconda guerra mondiale, è venuto in mente lui, **Wolfenstein!** Nulla da eccepire, avete ragione da vendere. La prima incarnazione di questa saga immortale è datata 1981: forse non tutti sanno che questo capolavoro vide la luce su Apple II. Il gioco, che prende il nome di **Castle Wolfenstein**, ha una grafica molto semplice e adotta una visuale dall'alto ma con rappresentazione dei personaggi di profilo. Ciò che dovremo fare è girare nel castello, composto da 64 stanze, alla ricerca di documenti segreti. In ogni stanza potremo trovare dei forzieri contenenti oggetti utili al proseguo del gioco, ma anche dei nemici pronti a farci fuori. Questi possono essere o soldati semplici, facili da abbattere, o le letali SS, dotate di giubbotto antiproiettile, più difficili da sconfiggere. Il seguito ufficiale arrivò nel 1984 con titolo **Beyond Castle Wolfenstein**. Fu inizialmente prodotto sempre su Apple e in seguito convertito anche su Atari 8 bit e Commodore 64. Qui la trama attinge ad un fatto storico ben preciso: l'attentato a Hitler del luglio 1944. Nel gioco dovremo infatti attraversare il labirintico bunker segreto dei Furher



**Wolfenstein 3D**

per giungere alla stanza dove egli tiene le sue riunioni, al fine di piazzare una bomba all'esterno di essa. Ma il vero boato questo titolo lo fece nel 1992 quando divenne uno sparatutto in prima persona per PC! **Chiamato Wolfenstein 3D**, ci vede alla guida del mitico William "B.J." Joseph Blazkowicz, un soldato statunitense che si trova prigioniero nell'oramai celebre castello. Il gioco consta in più livelli, ovviamente infestati da guardie armate e cani feroci. Il castello è più intricato del labirinto di Minosse, girarlo senza una mappa è veramente dura. In ogni livello vi sono stanze segrete ricolme di tesori, cibo, kit del pronto soccorso e diversi tipi di armi. Ciò che rese immortale questo capitolo fu il suo innovativo motore grafico e la visuale in prima persona. La tecnica del pseudo-3D era già stata utilizzata in un paio di titoli ma qui il livello si alzava decisamente, portando molti più dettagli e colori su schermo. Era qualcosa di decisamente pionieristico, lo scrolling era veloce ma innaturale e spesso causava senso di nausea o male agli occhi. Nei piani iniziali dei programmatori l'aspetto stealth avrebbe dovuto essere maggiormente presente, ma queste modalità più simulate furono abbandonate, visto che avrebbero reso il gioco meno dinamico e troppo complicato da gestire.



**Destroyer**



**Desert Fox**

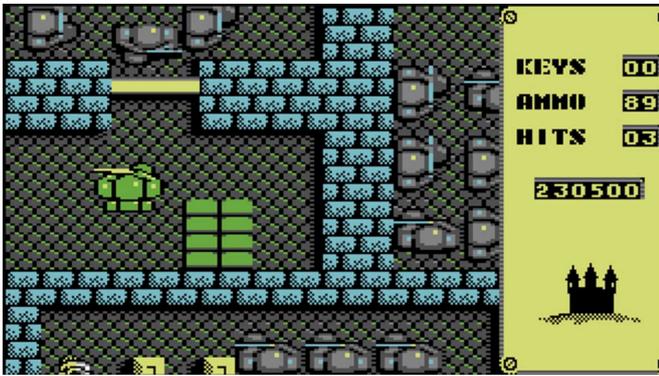
Anche il settore navale fu preso e analizzato per creare dei giochi notevoli. **Destroyer**, per esempio, è un gran bel simulatore, pubblicato nel 1986 dalla Epyx per Commodore 64, Apple II e i vari IBM compatibili. Nel gioco ci troveremo al comando di un cacciatorpediniere della Classe Fletcher, immersi in un contesto estratto dalla Seconda guerra mondiale, pur non facendo riferimento a nessun evento particolare. Non è un gioco semplice, visto che dovremo gestire sia la navigazione del mezzo che il combattimento, con tanto di puntamento manuale delle armi. Per farvi capire la difficoltà, dovremo muoverci fra 13 schermate differenti, ognuna rappresentante una postazione. Per cambiare postazione dovremo inserire un codice di due lettere, il che farà partire un caricamento del gioco. L'azione si svolge tempo reale, non ha importanza in quale schermata ci troviamo: ogni evento di una certa portata verrà segnalato tramite una barra dei messaggi. Un vero masterpiece non solo per quegli anni ma anche oggi giocarci è una sfida. Peccato per la versione Amiga non all'altezza: uscita nel 1988 è praticamente una trasposizione di quella per Commodore 64.

E che dire del sublime **Desert Fox**? Si tratta di un videogioco di strategia e azione, pubblicato nel 1985 per Commodore 64 e nel 1986 per Amstrad CPC. Questa volta ci spostiamo nel deserto, più precisamente nel Nordafrica. Siamo a bordo di un potente M4 Sherman degli Alleati chiamato "Lone Wolf", e dovremo cimentarci in molte missioni contro le forze del generale tedesco Erwin Rommel, detto "volpe del deserto". Nella modalità campagna il contesto è quello del 1941 in Libia: ci troviamo nella lingua di terra compresa tra Tobruch e Sollum e il nostro compito sarà quello di proteggere i depositi alleati dagli attacchi di Rommel. Sulla nostra mappa vedremo solo i depositi segnati da una bandiera, il nostro carro e una svastica ad indicare il generale tedesco. Le truppe nemiche non

sono segnalate, si dovrà usare l'astuzia tentando di intercettare le loro comunicazioni al fine di capire la loro posizione ed evitarle. Quando abbiamo la percezione di essere in guai seri potremo chiamare un attacco aereo per guadagnare tempo. Quando raggiungeremo un deposito sarà considerato salvo e noi riceveremo un pieno di energia. Sarà comunque facile imbatterci in qualche mezzo nemico, o in Rommel in persona visto che il suo carro è costantemente sulle nostre tracce. In questi casi si passerà in modalità combattimento, tutte con visuale in prima persona. Nulla è lasciato al caso, perfino la sequenza giorno e notte viene rispettata e vi assicuro che affrontare una battaglia col buio mette davvero ansia. Il Game Over arriverà se un deposito sarà conquistato dai tedeschi o se il nostro carro verrà distrutto.

Restando sul pianeta 8 bit fece molto scalpore nel 1986 **The Great Escape**. L'anno è il 1942 e dovremo controllare un prigioniero di guerra nel tentativo di organizzare la fuga da un campo di concentramento tedesco. L'ambiente di gioco è il campo di prigionia con visuale isometrica e scorrimento multidirezionale. Vi è solo un ingresso al campo, ultra-sorvegliato: il passaggio è consentito solo se in possesso dei documenti corretti. Gli altri lati del campo di prigionia danno su irte scogliere che si buttano in mare. L'azione parte dalla camerata, mostrata in schermata fissa, dalla quale parte la giornata: essa ha una routine quotidiana composta dall'appello, l'esercizio, i pasti e l'ora di andare a dormire. Questo ritmo è uguale per tutti i prigionieri, che seguiranno questo iter; se lasceremo il nostro personaggio fermo, dopo un po' di tempo si adeguerà alla routine. Ovviamente il campo è circondato da recinzioni e da cani da guardia, usati per pattugliare il perimetro. A questo aggiungiamo le guardie nelle torri di con tanto di proiettori, per osservare eventuali

**The Great Escape**



### Into the Eagle's Nest

prigionieri che cercano di scappare. In tutto questo contesto noi dobbiamo cercare di non farci beccare in giro per il campo in cerca degli oggetti che gli serviranno alla fuga. Una bandiera rappresenta il livello del morale del prigioniero: se va a zero termina la partita è persa. Il colore della medesima ci avvisa se siamo in una zona pericolosa (bandiera rossa) o tranquilla (bandiera verde). Potremo trasportare solo due oggetti alla volta, perciò si deve pensare bene cosa portarsi dietro ogni volta in base alle azioni che si vogliono intraprendere. Ad esempio, inoltrarsi nei cunicoli sotterranei senza luce è impresa quasi impossibile. Dovremo perciò nascondere in giro tutti gli oggetti trovati, i quali potranno esserci utili in seguito. Se ciò non fosse già abbastanza, le guardie potrebbero perquisirci ed arrestarci nelle zone di bandiera rossa, mentre il capitano del campo può farlo sempre e comunque. Sono sincero, ci ho giocato tantissimo ma non sono mai riuscito una sola volta ad evadere dal campo.

Un gioco che mi fece letteralmente impazzire fu il colossale **Into the Eagle's Nest**. Uscito nel 1987 per le piattaforme 8 bit e poco dopo anche per i primi 16 bit, il gioco riprende la dinamica di Gauntlet e la trasporta nella Seconda guerra mondiale. Il titolo è liberamente ispirato all'edificio realizzato sulla cima del Kehlstein, che si collega con la sottostante villa denominata "Berghof". L'edificio in questione non fu mai utilizzato per scopi militari, li si svolsero soltanto riunioni e incontri diplomatici. Ma la sua fama postuma e il nome altisonante, coniato dal giornalista britannico George Ward Price e chiaro riferimento all'antico simbolo della Germania, erano un'occasione troppo ghiotta per non farne un gioco. L'azione si svolge all'interno della fortezza, suddivisa in più piani, con visuale dall'alto e scorrimento multidirezionale. Gli ambienti sono zeppi di soldati tedeschi che ci possono danneggiare toccandoci, mentre noi abbiamo un mitra con massimo 99 colpi per difenderci. Dobbiamo liberare tre prigionieri

alleati, che sono le prime tre missioni del gioco: una volta salvato il soldato egli ci seguirà fino all'uscita del livello. La quarta missione è la più difficile, visto che si dovranno attivare le cariche di esplosivo necessarie per distruggere la fortezza e scappare per tempo. Per me fu un capolavoro assoluto!

In mezzo a tutto questo ben di dio non può mancare un gioco a tema sottomarino. In questa casella mi sento di incastonare un diamante di rara bellezza, ovvero **Silent Service** del 1985. Parliamo di un simulatore di sottomarino prodotto da Microprose per i maggiori home computer a 8 bit da Sid Meier, mentre nel 1986 uscì anche sui sistemi a 16-bit. Un gioco così realistico da creare alcuni problemi in Germania Ovest, qui intitolato Das U-Boot. L'ambientazione è l'oceano Pacifico e il periodo è la Seconda guerra mondiale. Il nome del gioco deriva dal soprannome dei sottomarini militari statunitensi stanziati nelle basi dell'Oceano Pacifico. Il gioco si ispira alle battaglie avvenute in pieno oceano tra sottomarini americani e le navi giapponesi: lo scopo è quello di affondare le flotte mercantili nipponiche affinché non riescano a rifornire le truppe del sol levante. Anche qui, come in Ace of Aces, la dinamica si snoda attraverso più schermate di visualizzazione: mappe, quadro comandi, interni, periscopio quando in immersione e torretta quando emerso. L'accuratezza è incredibile, con il meteo che cambia pregiudicando a volte la visibilità, e il ciclo giorno e notte, che ci permette di usare il buio per attacchi a sorpresa. Silent service è pazzesco, possiamo utilizzare le tattiche realmente usate nel conflitto, compresa la possibilità di scendere in profondità sul fondale, evitando così le bombe di profondità giapponesi. La cura dei dettagli è maniacale: le mappe sono realistiche in ogni dettaglio, comprese le profondità



### Silent Service





### Sniper Elite

marine. Le armi utilizzabili ricalcano fedelmente quelle reali, nonché le loro caratteristiche: ad esempio i siluri lasciano la scia nell'acqua e possono essere difettosi, con i problemi che ne derivano. I cacciatorpediniere giapponesi possono rilevare la presenza del sottomarino usando sonar o semplicemente osservando la superficie dell'oceano: se il sottomarino sta viaggiando a bassa profondità, esso lascerà una scia facile da individuare. In poche parole, un masterpiece che merita di essere riscoperto.

### Tempi moderni

E poi si giunge fino ai giorni nostri, dove Pc e Console sono sempre più potenti e permettono di realizzare videogames quasi cinematografici. Le saghe di Call of duty e Medal of Honor sono avvincenti e sempre più dettagliate. Vengono riproposti i fatti storici con un'accuratezza da lasciare senza fiato. Chi sta pensando allo sbarco in Normandia di Medal of Honor ci ha preso in pieno: atmosfere incredibili, ci si trova catapultati in mezzo all'azione frenetica e folle con il solo pensiero di portare a casa la pelle mentre dai bunker sulla spiaggia piovono tonnellate di piombo. Ma per me tra i giochi "nuovi" sul tema Seconda guerra mondiale ve ne è uno che reputo inarrivabile, unico, magnifico. Quasi perfetto. Parlo di **Sniper Élite**, uscito nel 2005. Il giocatore veste il ruolo di Karl Fairburne, un agente statunitense della Office of Strategic Services sotto sembianze tedesche. Il personaggio si inserisce nella Battaglia di Berlino del 1945, durante gli ultimi giorni del conflitto: Il 16 aprile 1945 l'Armata Rossa sferrò la sua ultima offensiva generale, con obiettivo Berlino. Le perdite furono altissime e lo sfondamento decisivo, ottenuto con la forza bruta di migliaia di carri armati, fu ottenuto solo il 20 aprile. In questo contesto si inserisce la trama del gioco, che ci

vede impegnati sotto mentite spoglie al fine di ottenere i dati sul programma nucleare militare tedesco prima che lo faccia l'Unione Sovietica. Durante tutto lo svolgimento del gioco avremo a che fare con diverse fazioni che si contendono il controllo della città: la resistenza tedesca, che ci viene in aiuto, il NKVD sovietico e le forze naziste. Durante il proseguimento della trama ci troveremo dinnanzi a personaggi storicamente esistiti, come Martin Bormann, figura estremamente importante del Reich. Egli sarà uno degli obiettivi che Karl dovrà eliminare! A differenza della totalità degli altri giochi più recenti sul genere, in Sniper Élite muoversi in modalità assolutamente stealth è l'unico modo di procedere nel gioco. Osservare bene la scena prima di muovere un passo fa capire come fosse terrificante la situazione nella realtà. Si passa la maggior parte del tempo strisciando o muovendosi in ginocchio in cerca di un riparo. Il realismo del sistema di puntamento è pazzesco: tenere il fiato, osservare stracci, bandiere o foglie sugli alberi per capire direzione e intensità del vento è la base per un tiro di precisione. Il punto in cui il proiettile colpisce il nemico sortisce diversi effetti, dal ferimento leggero che metterà all'erta il soldato al ferimento pesante. Ad esempio, un nemico ferito ad una gamba striscerà per cercare riparo o dare l'allarme. Un colpo alla testa è semplicemente one shot, one kill. Sniper Elite è stato premiato nel 2005 dalla The Independent Games Developers Association come "Best PC/Console Game", e non poteva essere altrimenti.

### The War is over

Come è finita la Seconda guerra mondiale lo sappiamo tutti: Il 6 agosto il bombardiere B-29 Enola Gay sganciò una bomba all'uranio (Little Boy) sulla città giapponese di Hiroshima. Tre quarti della città furono distrutti e 78.000 persone morirono all'istante. Tre giorni più tardi, il 9 agosto, il B-29 Bockscar sganciò una bomba al plutonio (Fat Man) sulla città di Nagasaki. Due quinti dell'abitato furono spazzati via e le vittime immediate furono 35.000. Ma molte migliaia di persone perirono nei giorni seguenti a causa delle gravi ustioni e dell'avvelenamento da radiazioni. **È davvero molto importante trasmettere e mantenere viva la memoria di questi tremendi fatti storici affinché non accadano mai più.**

E se oltretutto son riuscito a farvi riscoprire qualche titolone dimenticato... beh, allora anche per me è missione compiuta.

Passo e chiudo.





# Amiga: le 10 (+1 bonus) game intro più spettacolari

di Beppe Rinella

Bentornati a voi amiche e amici che, come me, scoprendo che i Commodores suonavano normalissimi strumenti musicali, anzichè meravigliosi Amiga, ci siete rimasti molto male!

Come noi tutti ben sappiamo, in sala giochi non c'era certo tempo da perdere, una volta inserito il gettone si entrava subito nel vivo dell'azione.

Non erano molti i titoli che prevedevano una piccola introduzione che raccontasse cosa stava succedendo, ma c'erano.

Così al volo mi vengono in mente Ghost'n Goblins, dove assistiamo impotenti al rapimento della nostra bella che dovremo liberare, o Double Dragon, dove in pratica succede la stessa cosa.

Si trattava comunque di pochi secondi e spesso con grafica "in game", nulla di minimamente cinematografico insomma. Alcuni titoli, più in là con il tempo, fecero qualcosa in più, come Final Fight ad esempio, dove era possibile vedere l'intro nei rari momenti in cui il cabinato era libero.

Le ragioni di tutto ciò erano ovvie, in sala giochi si giocava, non c'era tempo da perdere, la fila era lunga e i soldi da spendere tanti.

Con il gioco casalingo però la questione cambiava e molto, avevamo tutto il tempo di goderci il racconto della storia a cui avremmo preso parte, chi era il protagonista, cosa stava accadendo e perché.

La componente narrativa aumentò sempre più la sua importanza, con il tempo e grazie a console e personal computer sempre più potenti, l'intro divenne via via sempre più di impatto, in alcuni casi altamente spettacolare.



Another World

Tutto ciò per presentare la lista di questo mese:

## LE 10(+1 bonus) INTRO AMIGA PIU' SPETTACOLARI

Attenzione, con "spettacolari" non intendo unicamente l'aspetto estetico, che come vedremo in alcuni casi può non vantare effetti grandiosi, intendo tutto l'insieme...ma insomma, ci siamo capiti.

### ANOTHER WORLD

Ne ho abbondantemente parlato qualche numero fa, del gioco e, inevitabilmente, della sua intro.

La sequenza iniziale di Another World è semplicemente perfetta da qualunque punto di vista, se esistesse un manuale su come una storia dovrebbe essere introdotta, Another World avrebbe un capitolo dedicato.

Partendo dall'aspetto estetico, graficamente eccezionale, con un 3d meraviglioso e animazioni spettacolari grazie anche all'uso del rotoscope. Il sonoro non è da meno, pochi effetti ma azzeccatissimi (sono sicuro che molti di voi, se non tutti, sono in grado di "canticchiare" la sequenza dei tasti premuti dal protagonista per accedere al laboratorio). Ma soprattutto è la musica che contribuisce massicciamente alla resa finale, creando la suspense giusta e facendoci percepire fin da subito che qualcosa di grosso sta per accadere e non si tratta di nulla di buono. Ciò che, a mio avviso, rende magnifica l'introduzione di Another World è il farci vivere insieme al protagonista ciò che sta avvenendo, ignari quanto lui, non ci viene detto assolutamente niente.

Una totale mancanza di informazioni che non fa che accrescere il senso di spaesamento che ci accompagnerà poi nel corso della nostra avventura.

Mai prima di allora avevamo visto un incontro così ravvicinato tra videogioco e cinema, semplicemente un pezzo di storia.

### PREMIERE

Un gioco davvero delizioso Premiere, la presentazione che introduce il protagonista e ciò che sta per accadergli, lo è altrettanto.

Clutch, questo il nome del personaggio che saremo chiamati a guidare, è un montatore di pellicole



**Premiere**

cinematografiche. Durante il turno notturno si addormenta e le pellicole su cui sta lavorando e che dovranno essere pronte per la sera seguente, vengono rubate. La mattina, dopo essere stato svegliato da una telefonata del boss, scopre il furto e parte alla ricerca dei film perduti.

Tutto questo ci viene raccontato tramite un delizioso piccolo corto animato, con uno stile molto cinematografico e dunque perfettamente in linea con il tema principale del gioco. La sequenza animata altro non è che l'inizio del film "Premiere", con tanto di titoli di testa, tra cui compare il nome del protagonista che viene quindi presentato come un attore.

Tutto è realizzato divinamente, un cartone animato con tutti i crismi, disegnato e musicato alla perfezione. D'altronde Premiere fu sviluppato da Core Design, che in materia di intro animate e in stile cartoon, sapeva il fatto suo e lo aveva già ampiamente dimostrato con titoli come Chuck Rock, Heimdall o...ci arriviamo tra un po'.

Ogni qual volta decidevo di giocare Premiere, non credo di aver mai saltato l'intro, bellissima e che mi faceva entrare nel "mood" giusto per affrontare l'avventura insieme a Clutch.

### **CANNON FODDER**

Una delle intro più iconiche, per uno dei migliori titoli della storia di Amiga.

Quando si trattava di Sensible Software, chiunque sapeva di non doversi aspettare grafiche ed effetti spettacolari. Cannon Fodder e la sua presentazione non fecero eccezione, puntando tutto su ironia (troppa per alcuni, ma ci arrivo) e una canzone che una volta entrata in testa non ne uscirà mai più.

In questo caso l'introduzione non ha lo scopo di raccontare storia e personaggi del gioco, ma di presentarci il team che lo ha sviluppato.

Il tutto viene fatto tramite alcune foto, accompagnate da nome e personaggio "interpretato", dei singoli membri del gruppo, tutti vestiti con abiti militari. Come "STOO as Elvis" ad esempio, con foto di Elvis Presley anche lui in tenuta militare.

Visivamente divertente quanto per nulla spettacolare.

Ma è con la canzone che accompagna il tutto che i ragazzi sensibili compiono il capolavoro, "War never been so much fun" è una piccola grande perla, con il suo ritmo in levare, i fiati e la calda voce femminile che ci canta quanto la guerra non sia mai stata così divertente.

Composta da Jon Hare e il compianto Richard Joseph, non fu questa l'unica canzone che composero per Cannon Fodder, scrissero infatti anche "Narcissus", il tema che possiamo sentire quando vediamo la collina con le tombe dei caduti, struggente e meravigliosa, con al sassofono Ian Ritchie, che suonò tra gli altri con un certo Roger Waters. Venendo all'ironia da molti allora giudicata eccessiva, Cannon Fodder ebbe non pochi problemi in terra di Albione. Tutto a causa del papavero che vediamo nell'intro (e che compariva inizialmente sulla copertina del gioco), che non è un fiore qualunque, viene infatti usato come simbolo della Royal British Legion, un'organizzazione benefica che sostiene i veterani di guerra. Se poi aggiungiamo il titolo del tema principale e il tono generale del gioco, dove ci si diverte bellamente uccidendo soldati qua e là, la polemica è servita. Qualcosa di più di una polemica in realtà, si andò infatti per vie legali ma tutto si risolse con il pagamento, da parte di Sensible, di 500 sterline per i diritti d'uso del papavero. Decisamente un bel guadagno per il team di sviluppo inglese, data la grande quantità di pubblicità gratuita che tutta la faccenda fece al gioco, che in effetti (ma non solo per questo motivo, lo sappiamo)



Designed by  
Sensible Software.

**Cannon Fodder**



**Superfrog**

vendette decisamente bene.

Ed ecco spiegato il perchè del disclaimer con cui si apre l'intro.

Ultima nota, la coppia Jon Hare/Richard Joseph ci prese evidentemente gusto, insieme composero anche i temi di Cannon Fodder 2, Sensible Golf e Sensible World of Soccer.

### **SUPERFROG**

Parlando delle migliori intro mai viste su Amiga, non è davvero possibile non menzionare Superfrog.

La storia che racconta non è particolarmente originale e neanche vuole esserlo: il classico principe e la sua altrettanto classica principessa infastidiscono la (anche lei) classica strega cattiva, che proprio non tollera tante smancerie. Decide così di rapire la principessa, non prima di aver trasformato il principe in ranocchio, lasciandolo così triste e sconcolato in riva al fiume. Proprio il fiume porterà al nostro (quasi) eroe una bottiglia di Lucozade (complimenti agli zozzoni che ce l'hanno buttata eh!), bibita da noi sconosciuta nonché sponsor del gioco e che berrà senza pensarci troppo, trasformandosi così in Superfrog, con immancabile mantello rosso da vero supereroe.

Tutto ciò ci viene raccontato attraverso un sublime corto animato, realizzato magistralmente da qualunque punto di vista. Colorato, animato splendidamente, allegro e con un paio di gag che strappano un bel sorrisone, come il tecnico che dimentica di far partire il ventilatore, non facendo così svolazzare il mantello del nostro eroe in posa plastica (rivelando quindi come tutto ciò che vediamo sia un semplice set cinematografico). O la figuraccia rimediata dalla nostra ranocchia quando tenta di spiccare il volo, che fa tanto Ralph Supermaxieroe.

L'autore di tutto ciò fu il fumettista statunitense Eric W. Schwartz, con il suo stile decisamente ispirato ai Looney Tunes, che tra le altre cose creò (utilizzando proprio un Amiga) il personaggio "Amy the Squirrel", una femmina

di scoiattolo antropomorfo che raggiunse un livello di popolarità tale da essere stata "adottata" come mascotte non ufficiale di Amiga.

Superfrog, a torto o ragione, veniva (e viene) spesso paragonato a Sonic e Mario, autentici capolavori oltre che metri di paragone per qualunque platform dell'epoca. Nessuno di questi due titoli poteva però vantare una presentazione animata di siffatta beltade.

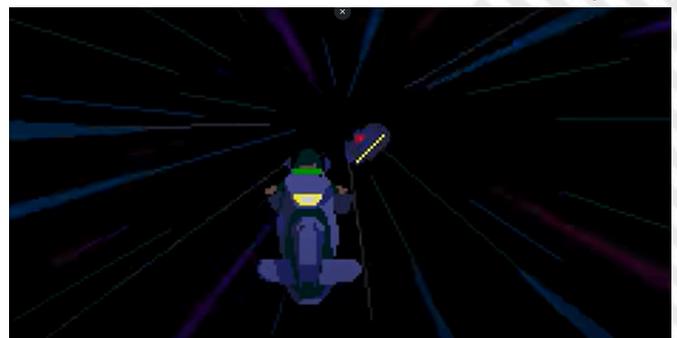
### **FLASHBACK**

Un titolo a dir poco magnifico, che deve davvero tanto al suo predecessore Another World uscito solo un anno prima, predecessore "solo" spirituale però, considerando che i due capolavori non sono minimamente collegati tra loro. L'eredità lasciata da Another World, raccolta a piene mani da Flashback, è già ben visibile nella sequenza che introduce la storia.

Partendo dalla realizzazione grafica sopraffina, troviamo anche qui un massiccio uso di animazioni realizzate con la tecnica "rotoscope", utilizzata anche per le animazioni in-game. Partita l'intro assistiamo ad un inseguimento, un uomo sta scappando (che deve essere il protagonista perchè sì, il protagonista è sempre quello che fugge) inseguito da brutti ceffi armati di pistola laser, che non esitano ad usare contro di lui. Il fuggitivo sale a bordo di un'avveniristica moto volante e prosegue la sua fuga, i cattivi di turno però non perdono tempo, salgono a bordo della loro astronave decisamente più performante, raggiungono in brevissimo tempo il malcapitato e lo abbattano senza troppi complimenti a colpi di laser, facendolo precipitare nel mezzo di una foresta.

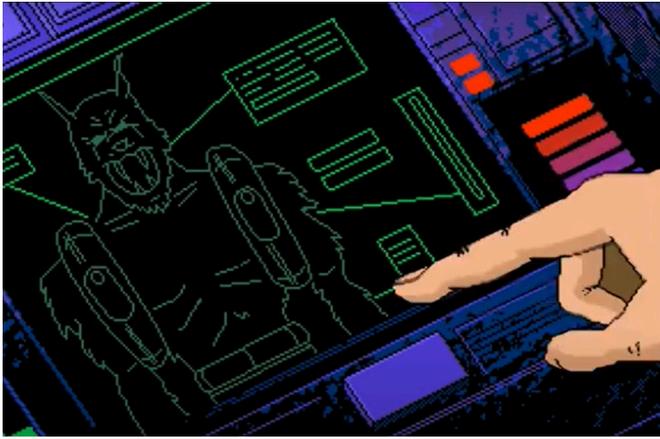
Così come avviene nell'introduzione di Another World non sappiamo nulla di ciò che sta avvenendo davanti ai nostri occhi, finita la sequenza animata abbiamo più dubbi che certezze.

Chi è quell'uomo? Perchè scappa e da chi? Dove si trovano? Le risposte arriveranno solo impersonando l'uomo misterioso e portando a termine questa fantastica avventura. Un'introduzione emozionante, carica di azione e pathos,



**Flashback**





### Wolfchild

grazie anche all'ottima musica che accompagna il tutto, davvero una piccola gemma.

L'aggettivo "piccola" non è casuale, la durata della sequenza animata è infatti di ben 40 secondi esatti, più che sufficienti a farci venire una gran voglia di impugnare il joystick e scoprire cosa diavolo sta succedendo.

### WOLFCHILD

Anche in questo caso Core Design dimostra di sapere come si fa, in materia di intro animate.

Kal Morrov, scienziato di fama mondiale, viene rapito dall'organizzazione criminale Chimera, che vuole sfruttare le incredibili scoperte del dottore, che permettono la creazione di ibridi tra uomini ed animali. Inutile dire che tutto ciò ha lo scopo di formare un esercito di soldati dotati di forza e capacità straordinarie.

Il figlio del dottore Saul, protagonista di questo grandissimo titolo, decide così di affrontare la temibile Chimera e liberare il padre, ma non può farcela come semplice uomo (per quanto decisamente fisicato), per affrontare una missione di tale portata decide così di attivare il progetto Wolfchild.

L'intro ci mostra il protagonista turbato e decisamente arrabbiato, solitario nella sua casa in cima a una montagna, che sperimenta su di sé le scoperte del padre, trasformandosi in un temibile licantropo.

La presentazione è di altissimo livello, graficamente ottima, il desiderio di vendetta e rivalsa del nostro eroe è palpabile, avvenuta l'esaltante trasformazione non aspettiamo altro che impersonare il nostro eroe e prendere a cazzotti chiunque ostacoli il nostro percorso. Non hanno idea di chi si sono messi contro.

Una gran bella "origin story" degna del più classico dei supereroi. Un uomo spaventosamente arrabbiato e affamato di vendetta che si trasforma in un fortissimo

uomo-lupo, ditemi voi cosa può esserci di altrettanto esaltante!

### EPIC

Volendo riassumere in una sola frase la storia di Epic, possiamo dire che racconta la più grande migrazione di massa della storia (dei videogiochi).

La stella che illumina e scalda il nostro pianeta è destinato a morire a breve, trasformandosi in una devastante Supernova. La sola speranza di sopravvivenza per la nostra civiltà è quella di colonizzare l'unico pianeta giudicato adatto ad ospitarci, Ulysses 7.

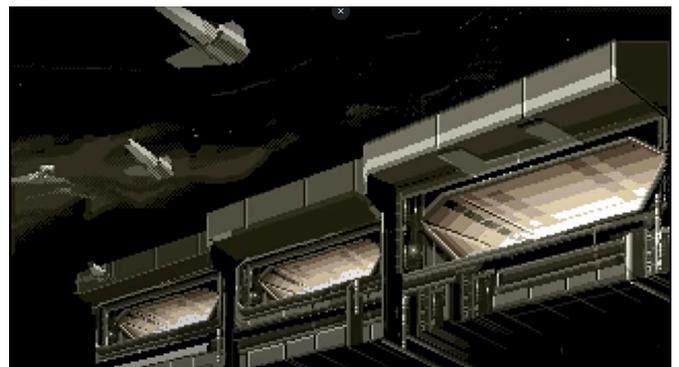
Il problema è che per arrivarci dovremo attraversare il territorio dell'impero Rexxon, civiltà aliena sempre ben disposta a scatenare una guerra e con cui ogni tentativo diplomatico per scongiurare una sanguinosa battaglia, è miseramente fallito.

L'unica speranza di sopravvivenza alla netta superiorità del malvagio impero, è costituita da tre astronavi costruite utilizzando l'Epical, un materiale particolarmente resistente. Una di queste sarà quella che saremo chiamati a pilotare (le altre due scoteranno la popolazione migrante) per combattere il malvagio nemico.

Nell'intro animata ci viene raccontato tutto ciò mostrandoci il grande esodo, un momento drammatico sottolineato dall'ottima musica, epica e minacciosa come il viaggio che il nostro popolo sta affrontando.

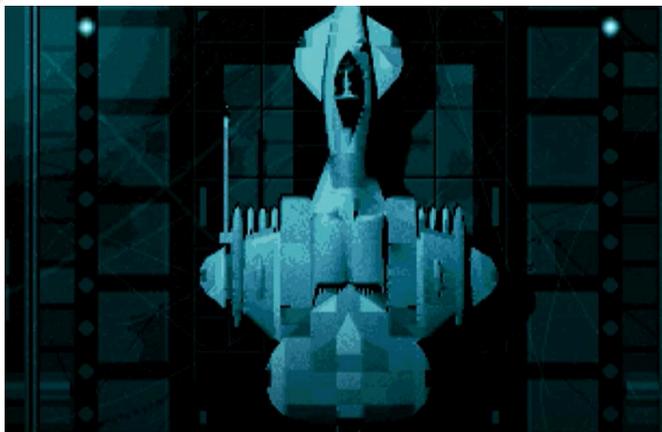
Graficamente è davvero una gioia per gli occhi, una combinazione di paesaggi statici, ottimamente realizzati, grazie ai quali possiamo ammirare il nostro pianeta di origine oltre a quelli che incontriamo lungo il nostro viaggio. Paesaggi che fanno da sfondo alle imponenti flotte in movimento, realizzate con un ottimo 3d per l'epoca (siamo nel '92) davvero notevole.

La sequenza di Epic è tutto ciò che ogni appassionato di Star Trek, Star Wars, galassie lontane e qualunque altra cosa riguardi battaglie spaziali desiderava allora come oggi, e io sono decisamente tra loro.



Epic





**Aquaventura**

## AQUAVENTURA

La presentazione di Aquaventura è probabilmente il miglior esempio di specchietto per le allodole della storia dei giochi Amiga. La sua incredibile qualità fu grande tanto quanto la delusione per il gioco vero e proprio.

Ma stiamo parlando di intro senza preoccuparci di ciò che succede dopo, e l'intro di Aquaventura è forse tra le più spettacolari mai viste sul nostro amato home computer. La storia di Aquaventura, che peraltro non viene raccontata attraverso la sequenza animata, è molto semplice: il nostro amato pianeta è devastato da una guerra nucleare, i pochi sopravvissuti hanno formato otto colonie sottomarine, fino a quando una razza aliena e ostile decide di invadere il pianeta, occupando tutte le colonie. I pochi sopravvissuti vivono in un'astronave in orbita nello spazio, da qui partiremo a bordo della nostra navicella per fare ritorno sulla Terra, nel tentativo di riconquistare le colonie terrestri. E' da questo momento che parte l'intro, vediamo la partenza della nostra piccola astronave con a bordo Josh Aldrin, il nostro alter ego.

Come già accennato, la sequenza animata è tecnicamente impressionante, creata con un ray-tracing assolutamente sbalorditivo per l'epoca. Ricordiamo che Aquaventura uscì nel 1992 ma l'intro iniziò a circolare già ben prima, il gioco infatti fu annunciato quattro anni prima della sua uscita, un periodo di tempo enorme per gli standard dell'epoca.

Nel corso della presentazione possiamo ammirare la nostra piccola astronave, con un design che ancora oggi fa una gran bella figura. All'interno, come detto, vediamo il pilota che con un cenno della testa dà l'Ok, i motori esplodono tutta la loro potenza e si parte a gran velocità verso lo spazio profondo, non prima di aver letteralmente fatto evaporare un tizio che si trovava decisamente troppo vicino alla pista di decollo.

La sequenza è ansiogena, cupa e opprimente. E' tutto

molto scuro e non c'è alcuna musica in sottofondo, solo qualche bip, il rombo dei motori ovattati (lo sentiamo dall'interno della navicella) e l'esplosione finale. Si perchè una volta partiti, vedremo alle nostre spalle l'astronave "madre" allontanarsi ed esplodere poco dopo, rendendoci così l'ultimo essere umano sopravvissuto.

Insomma Aquaventura vanta una presentazione di altissimo livello, graficamente superba, che crea alla perfezione l'atmosfera di angoscia che si può provare poco prima della partenza per una missione disperata.

Peccato che il gioco non sia stato poi minimamente all'altezza.

## SHADOW OF THE BEAST 2

Mancava ancora la visione di qualcosa di decisamente tendente all'horror, vero?

Recuperiamo subito con una delle intro che ha fatto storia su Amiga, quella del secondo capitolo di Shadow of the beast. Un titolo a mio avviso quasi insopportabile, di una difficoltà totalmente insensata, ma è indubbio che la saga sia tra le più famose mai uscite per il nostro amato Commodore.

Zelek, negromante servo di Maletoth, ossia il villain sconfitto nel primo capitolo, rapisce la sorella del protagonista, per poterla poi utilizzare in non si sa quale rituale, riportando così il suo padrone alla sua forma fisica. Ciò a cui assistiamo è il momento del rapimento della povera infante, vediamo la casa sperduta nel nulla dall'esterno, Zelek che la osserva da lontano e che grazie alla sua magia si trasforma in un mostro alato. Dall'esterno dell'abitazione sentiamo il pianto di una bambina, Zelek piomba sul tetto e, dall'interno della casa vediamo la sua grande mano che, dopo aver distrutto il tetto, rapisce la pargola, tra le urla strazianti della madre.

Graficamente la sequenza animata è davvero notevole, l'atmosfera cupa e orrificata viene resa alla perfezione, l'ambientazione esterna per nulla rassicurante e desolante, la casa spoglia e solitaria.

Il sonoro fornisce a tutto ciò un grande contributo, i tuoni



**Shadow of the Beast 2**





### Syndicate

che rimbombano, il pianto della bambina, le urla disperate della donna. Funziona tutto alla perfezione.

L'intro di Shadow of the beast 2 è un piccolo capolavoro, soprattutto pensando all'anno della sua uscita.

Purtroppo il gioco non era umanamente sopportabile, terrificante anche lui, ma in questo caso non è certo un complimento.

### SYNDICATE

Chiunque abbia amato e ami ancora Blade Runner (presente!), l'estetica cyberpunk e futuri distopici, non poteva non adorare profondamente Syndicate, autentico capolavoro partorito dalle geniali menti di Bullfrog.

In un futuro, si spera, molto lontano il mondo è suddiviso in tre grandi aree, ognuna delle quali controllata da una grande corporazione. Lo scopo di ogni corporazione è quello di prevalere sulle altre e riuscire così a governare l'intero pianeta.

Ognuna di esse dispone del proprio esercito di cyborg mentalmente controllati, a cui vengono fatte eseguire le missioni necessarie al raggiungimento dello scopo finale. Scelta la corporazione a cui unirci, partiremo alla conquista del mondo senza scrupolo alcuno in questo magistrale strategico in tempo reale.

L'intro che ci introduce Syndicate non narra assolutamente nulla di tutto ciò, il tutto veniva raccontato nel manuale/libro del gioco, che ovviamente il sottoscritto non possedeva. Ciò che la presentazione fa è farci subito respirare l'atmosfera del gioco, opprimente e oscura come ogni futuro distopico che si rispetti.

Vediamo macchine fluttuanti, un enorme veicolo volante e un ignaro passante (così almeno sembra) che viene rapito, condotto al quartier generale di una delle corporazioni e a quel punto assistiamo a cosa significa essere forzatamente arruolato. Il malcapitato viene posizionato all'interno di un grande macchinario denominato Leonardo, la cui funzione è quella di innestare al povero uomo arti

biomeccanici e rendere la sua mente controllabile.

Tornando all'esterno, nel punto in cui l'uomo è stato rapito, lo ritroviamo lì, pronto ad eseguire gli ordini che gli verranno imposti.

La presentazione di Syndicate è graficamente davvero ottima, il design di veicoli e ambienti azzeccatissima, le animazioni semplicemente impeccabili. Ottimo anche l'uso dei colori, seppia per gli esterni, freddo e metallico blu per gli interni. La musica poi contribuisce a creare il senso di oppressione e tirannia in cui la popolazione mondiale è costretta a vivere.

Presentazione davvero spettacolare e carica di atmosfera, che introduce degnamente un titolo assolutamente magnifico.

### AGONY (BONUS)

Perché bonus? Beh, perché Agony non ha alcuna introduzione animata.

Allora cosa ci fa qui? La risposta è molto semplice: la musica.

Un tema musicale meraviglioso di oltre

quattro minuti e composto da Tim Wright, accompagnato da una splendida illustrazione realizzata da Frank Sauer (così come tutte le splendide immagini che introducono ogni livello del gioco), ci introduce nel fantastico mondo di Agony. Pianoforte e archi creano una melodia struggente, intima e sublime.

Non voglio scrivere oltre, non sarei in grado di descrivere tanta bellezza, ma sono sicuro che chiunque tra di voi, fratelli e sorelle Amighisti/e, sa perfettamente di cosa parlo. Ma se così non fosse, è indispensabile recuperare. Pura arte e meraviglia, scusate se è poco.

Ed eccoci arrivati alla fine di questa lista, ho scelto queste dieci magnifiche intro ma sono tante quelle "escluse" e che meritano di essere citate, magari in una futura seconda lista delle intro Amiga più spettacolari.

E le vostre preferite quali sono?

Venite a trovarci sui nostri canali social e diteci la vostra lista!

Un saluto a tutti voi e AMIGA FOREVER!



Agony





# Talent Scout

di Carlo Nithaiah Del Mar Pirazzini

Nasce una nuova rubrica per scoprire e parlare del numeroso gruppo di sviluppatori, grafici, musicisti italiani e non solo che stanno facendo tantissimi ottimi lavori per i nostri retro-computer/retro-console. Oggi vi presentiamo **Andrea Gasparrini** e la sua **GAMECAST Entertainment**.

Mi presento, il mio nome è Andrea Gasparrini e sono un appassionato sviluppatore indipendente e retrocomputerista. Ho sempre seguito lo standard MSX/MSX2, dagli anni 80 fino ad oggi, prima su MSX 1 e MSX 2 ed in seguito su MSX 2 Turbo – R e MSX 2+.

Dalla mia grande passione per i videogiochi sono diventato un piccolo sviluppatore indie per MSX/MSX2 e lavoro con Basic, Turbo Basic, Nestor Basic e in Assembler Zilog Z80. Però a casa possiedo anche alcune macchine 8 Bit Commodore e attualmente tre MSX 1.

In passato ho partecipato a molti concorsi della scena MSX come MSXDEV e anche alla MSXDEV COMPO.



**BARUKO – MSX1: Per la MSXDEV COMPO 2012**

Il nome del mio team è GameCast Entertainment, anche se precedentemente è possibile trovarmi con il nome AGSoftware (Andrea Gasparrini Software).

Anche questo 2022 parteciperò alla MSXDEV '22 con il mio nuovo videogioco spaziale chiamato DEFENCE, anche se credo di riuscire ad infilare qualche altro titolo.



**Il gioco DEFENCE per MSX in gara alla MSXDEV '22**

Nel 2021 ho partecipato con il titolo Pippo la Scopa Ficcanaso.



**PIPPA LA SCOPA FICCANASO – MSXDev '21**

Tra i tanti lavori che sto facendo e che ho fatto, nel 2017 tentai la pubblicazione di una rivista in pdf online che avevo chiamato DATAPRINT Magazine.

È possibile trovarla qui: <https://dataprint.altervista.org/>

La mia passione da programmatore nasce grazie alle riviste LIST degli anni 80 edite dalla Edicom che mi fecero innamorare del Basic.

Un saluto e grazie ancora.

Andrea Gasparrini

## Siti di riferimento

### Game Cast Entertainment:

<http://gamecast.altervista.org/index.php>

### MsxDev '22:

<http://www.msxdev.org/>

### DATAPRINT Magazine:

<https://dataprint.altervista.org/>





NEW GAME

# RANDOOM

È possibile che un gioco composto quasi esclusivamente da quadratini grossi come cursori possa avere senso? E soprattutto essere divertente?

Se parliamo di Commodore 64 e di Randoom la risposta è sì. Lo diciamo da un pezzo che questa rinascita delle piattaforme ad otto (e sedici) bit è forse la cosa più grande che potesse capitare al mercato videoludico. I titoli che stanno uscendo per le nostre retro-macchine da gioco sono sempre più incredibili, ma ancora di più lo sono le direzioni che prendono. Il gioco di cui andiamo a parlare oggi sceglie di prendere una direzione letteralmente casuale, come si evince dal nome stesso, e fa di questa caratteristica il suo punto di forza.

Il concetto è riassumibile in poche righe: ci troveremo ad inseguire dei cubi attraverso 31 schermate di gioco, facendo attenzione a prendere solo quello del colore richiesto, pena la perdita di una vita. Scritto così forse non dice molto, ma ancora più spiazzante è quando si comincia a giocare. Ad una musica assolutamente da premio Oscar fa da contraltare una grafica che definire semplicistica è un eufemismo. Sono rimasto fermo una trentina di secondi in una schermata fissa a vedere dei quadrati colorati muoversi: stavo cercando il mio personaggio. Decido di muovere il Joystick e vedo muoversi un altro quadrato che era fermo in mezzo allo schermo: cado in un buco, esco dalla parte alta dello schermo, cado di nuovo, mi scontro con un quadrato e muoio. Ricomincia il tutto da capo, ricado, risbatto e muoio. Eccomi nuovamente al principio, ma stavolta pigio il pulsante di fuoco e vedo il mio cubo saltare. Sto fermo e, analizzando la schermata di gioco, capisco che devo prendere i cubi che scappano secondo un ordine preciso che appare nella sezione dedicata alle info di gioco. In quel mentre mi balza in mente un nome che definire storico è poco: Bubble Bobble. Effettivamente il colpo d'occhio è quello: schermata di gioco vera e propria e a destra le informazioni (punteggio, vite e via discorrendo). Anche la dinamica,

semplificata, ci può stare: ci muoviamo su piattaforme, saltiamo e dobbiamo catturare dei nemici... mancano solo le bolle. Nulla di più sbagliato miei cari lettori, perché una volta superato l'impatto con la grafica basica ci si accorge di un dettaglio che sarà il punto di forza del titolo in questione: i cubi nemici non seguono una traiettoria prestabilita. Mi sono reso presto conto che ad ogni mia morte il livello ricominciava, ma loro si muovevano sempre in maniera differente. Salta così in aria la vecchia teoria che "si impara lo schema dei nemici e si prosegue": qui dobbiamo letteralmente improvvisare. Ci troveremo ad inseguire, a fuggire, a saltare e a cercare il nostro obiettivo sempre in maniera differente. Si fa in fretta a prendere la mano e a morire meno sovente ma non esiste una regola, dipende da come si muoveranno i cubi colorati e in base a questo il medesimo livello può risultare una passeggiata o un vero inferno.

La casualità, ecco il colpo di genio assoluto che ci farà restare letteralmente incollati a questo gioco per un sacco di tempo. Se riusciamo a catturare dei nemici in rapida sequenza avremo dei punti bonus e, raggiunti i 100 punti, una vita extra che non fa mai male. Un neo glielo voglio trovare: il primo e il penultimo cubo sono cromaticamente molto simili e talvolta mi sono trovato a catturare quello sbagliato. Ma a parte questo abbiamo tra le mani, a costo zero, un gioco pazzesco, un arcade platform che ci conferma come una buona e geniale idea sia la base per ottenere un grande gioco.

La grafica semplice, alla fine, risulta efficace e non fine a se stessa, mentre la traccia audio è veramente killer, una delle migliori prodotte dal SID negli ultimi anni. Chiudo gli occhi un attimo e immagino Randoom in una sala giochi nel 1986, messo in un accattivante cabinato dedicato: sarebbe stato un successo assicurato, magari con la possibilità di avere un 2 player game!

di **Mic the Biker Novarina**

Anno: 2021

Codice: Dozznar of Picaro Games

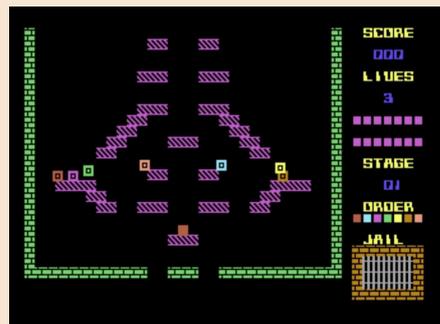
Musica: Narcisound of Onslaught

Grafica: Toni Galvez of Batman Group

Publisher: Picaro Games

Genere: Platform

Piattaforma: Commodore 64



## GIUDIZIO FINALE

### » Giocabilità 88%

Poche partite per capire cosa fare e come farlo, all'inizio morire è facile ma si impara in fretta come procedere.

### » Longevità 98%

Ogni livello non sarà mai giocato allo stesso modo, il fattore casualità rende questo titolo potenzialmente infinito. Non ve ne staccherete più!





NEW GAME

# RANDOMINER

Anno: 2022

Sviluppatore: Thomas Ilg

Genere: Puzzle

Piattaforma: Atari ST

Sito web: <https://www.hd-videofilm.com/randominer/>

Il Sistema di gioco è quello del famoso "Campo Minato" di Windows ma portato in isometrico e con diverse aggiunte.

Bisogna fare clic per aprire un riquadro. Il numero di tessera ci dirà quante mine sono presenti attorno a noi. Dovremo usare la logica, il cervello e spesso una alta dose di fortuna per determinare dove si nascondono le mine.

A questo si aggiungono alcuni aspetti ulteriori come ad esempio i punti bonus in più per le giocate veloci che mette un po' "sale" alla sfida.

Il gioco è eseguibile su tutte le macchine con almeno 1 mb di Ram. Si carica molto velocemente e gira senza problemi anche in emulazione.

È stato localizzato in numerose lingue, tra queste abbiamo anche l'italiano curato dal caro amico **Filippo Santellocco** del gruppo **Atari World Italia**.

E' un gioco ben sviluppato e solido, il classico titolo da caricare per passare qualche minuto piacevole nelle pause dalla vita reale.

Consigliatissimo.

di **Giampaolo Moraschi**



## GIUDIZIO FINALE



### » Giocabilità 80%

Uno dei puzzle game più giocati della storia in chiave moderna.

### » Longevità 80%

Ogni tanto lo caricherete. Tra una pausa e l'altra. Io l'ho trovato ben fatto e rilassante.





**NEW GAME**

# GREEN BERET

Anno: 2022

Editore: Konami

Genere: Shoot'em up

Piattaforma: Amiga

Sito web: <https://danteretrodev.itch.io/greenberet>

Un altro classico degli anni 80 arriva su Amiga. Ci ha messo il suo tempo, ha dovuto aspettare l'arrivo dello Scorpion Engine, ma finalmente è giunto.

Questi ultimi due anni sono stati ricchi di conversioni di titoli storici su Amiga, come ad esempio Yie-Ar Kung Fu o Metal Gear, ed è il momento di rispolverare un altro classico da sala giochi, ovvero Green Beret o, com'era conosciuto in USA, Rush'n Attack.

Lo gioco è stato creato nel momento storico dove il mondo era diviso in due (più o meno come ora, ma in modo più deciso). Noi vestiremo i panni di un eroico soldato dei corpi dei Berretti Verdi che ha il compito di liberare degli ostaggi presenti in alcuni campi di prigionia di un fantomatico stato "canaglia".

Per fare tutto ciò saremo armati di un incredibile pugnale affilato e solo tre vite. Comodo vero? Perché tutto il resto dell'armamentario dovremo procurarcelo noi con le nostre sole forze.

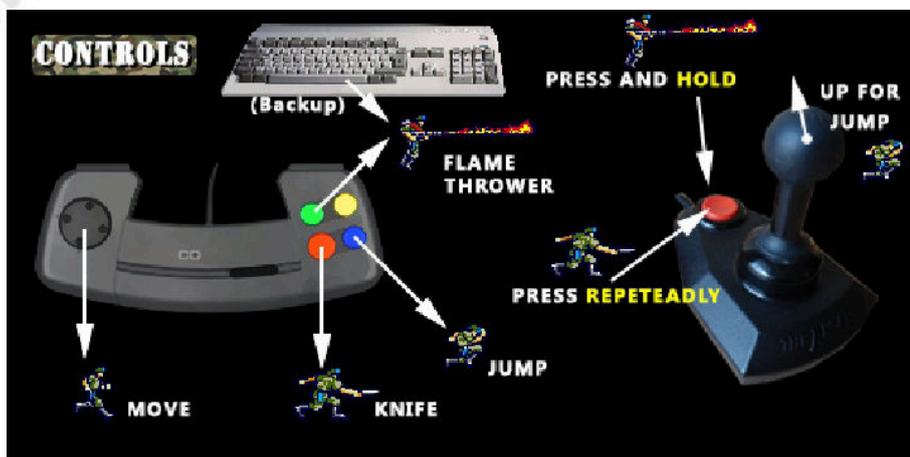
I nemici ovviamente cercheranno di farci la pelle in tutti i modi possibili. Ci occorreranno nervi salti e una forte memoria per riuscire ad arrivare alla fine del gioco.

È un gioco dinamico e difficile come



è uno dei migliori arcade dei favolosi anni 80. Infatti al tempo riscosse un successo incredibile e furono tantissime le





## GIUDIZIO FINALE

### » Giocabilità 85%

C'è tutto. Armi, livelli, nemici e ... imprecazioni!

### » Longevità 85%

Ha il feeling del gioco da sala giochi e quel suo modo di coinvolgerti sempre di più, partita dopo partita.

conversioni per gli home computer.

Le migliori versioni furono quella per Commodore 64, seguita da quella per ZX Spectrum mentre su console la versione NES fu un grande successo grazie ad una aggiunta di nuovi livelli e alcuni piccoli bonus in più.

Ma su Amiga? Non c'era nulla, oppure c'era Fire Force ma non era l'originale.

Nel 2022 tutto questo è risolto con una bella versione realizzata con lo Scorpion Engine. Sono presenti i 4 livelli del gioco originale con tutti i boss di fine livello, tutti i nemici, le tre armi secondarie della sala giochi e può anche essere giocato con il gamepad a due pulsanti oltre al canonico joystick.

Il gioco ha una bella colonna sonora rifatta dalla versione NES e gira su tutte le macchine dotate di 1 Mb di Ram.

Un titolo dall'aspetto identico all'originale, come è identico anche il gameplay e l'approccio al gioco.

Green Beret forse non piacerà alle nuove generazioni, ma tutti quelli che lo hanno amato in sala giochi lo apprezzeranno molto.

di **Roberto Del Mar Pirazzini**





**NEW GAME**

# METAL DRAGON

Anno: 2022

Sviluppatore: KAI MAGAZINE SOFTWARE

Genere: Shoot'em up

Piattaforma: Sega MegaDrive

Sito web: <https://kai-magazine-software.fwscart.com/>

Il Megadrive è una console incredibile! Sta godendo di una seconda giovinezza grazie ad una nutrita schiera di sviluppatori, grafici e programmatori. Negli ultimi anni abbiamo visto l'uscita di titoli in dose massiccia e quasi tutti di buonissima fattura.



Questo Metal Dragon è un titolo creato dal collettivo KAI MAGAZINE SOFTWARE, che lo aveva precedentemente rilasciato su MSX V9990. Il team è guidato da Oscar Kenneth Albero e ha la sua base in Spagna, terra che ci ha dato davvero tanto sul fronte retrogaming.

E' un grande viaggio nel tempo. Un viaggio tra i titoli mitici del primo periodo della console che ci riporta in un era "machista" e ricorda titoli come Mercs o Dynamite Duke. Titoli che ci trasformavano in "ragazzi duri" che da soli affrontavano interi eserciti per liberare il mondo.

E' chiaro fin dall'inizio che questo titolo per Megadrive è un esercizio di nostalgia, ma è anche una parodia di altri titoli, in particolare della serie Metal Gear, a cui lancia senza pietà molte provocazioni.

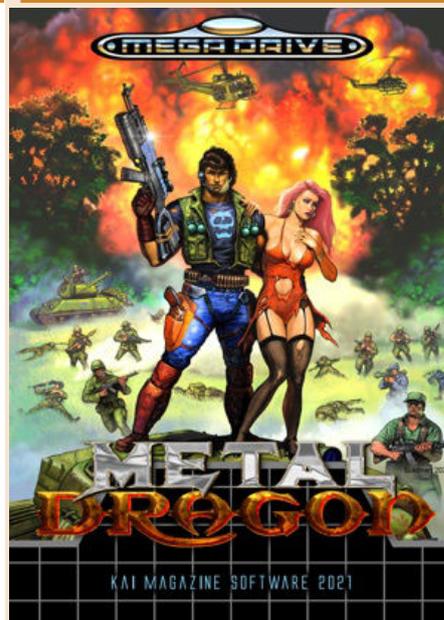
Il suo sviluppo ricorda

irrimediabilmente il già citato Mercs o Commando.

Alla guida del eroe che dà il nome al gioco, dovremo avanzare attraverso il livello mentre centinaia di soldati nemici ci spareranno contro qualsiasi tipo di arma.

Ovviamente dobbiamo sparargli per eliminarli, ma l'importante è avanzare fino alla fine dello schermo per passare all'area successiva. In altre parole: tranne che nei duelli contro i boss finali, non è strettamente necessario eliminare nessuno... Anche se la situazione ci "invita" costantemente a sparare.

Si partirà con un'arma con munizioni illimitate, ma dal danno bassissimo. Fortunatamente, eliminando alcuni nemici potremo ottenere un bellissimo e potentissimo arsenale che va dal lanciafiamme al mitra, dal bazooka





al fucile da cecchino.

Tutte queste armi hanno munizioni limitate e lo stesso vale per le granate, la cui quantità aumenterà solo trovando alcune scatole nei livelli. Le granate sono fondamentali per eliminare alcuni nemici particolari come i robot o i carri armati che compaiono.

Riservare le nostre armi migliori per i nemici più difficili è una buonissima idea, anche se giocando a difficoltà normale non avrete problemi di stoccaggio munizioni in generale (grossa pecca).

Per ripristinare i nostri danni corporei potremo raccogliere i kit di pronto soccorso che si trovano nei livelli per ripristinare la salute, anche se è sempre possibile resistere a diversi colpi senza morire.

Non abbiamo un numero di vite, ma una barra della salute che, se ridotta al minimo, porterà al Game Over. Un comodo sistema di password e i numerosi continue ci permetteranno di poter proseguire l'avventura senza troppi patemi e senza una reale sfida. In questo caso avrei preferito meno continue per rendere la longevità più alta.

Il gioco è comunque molto lungo e richiede impegno e molta memoria. Sono presenti quattro livelli di difficoltà ma solo il livello Hardcore rappresenta la sfida reale.

Il gioco presenta anche due finali. Uno molto catastrofico e uno con lieto fine. Manca la possibilità di giocare in doppio e devo ammettere che è un peccato per questo genere di giochi. In Mercs

era una possibilità meravigliosa.

A livello grafico è molto interessante. Ottimo uso del colore e del design dei nemici come della qualità del dettaglio dei fondali. Ogni tanto assistiamo a qualche sfarfallio degli sprites ma non compromette la visione di gioco.

Colonna sonora con ottimi effetti digitalizzati ma dalla musica di accompagnamento un po' piatta.

Pezzo forte è l'umorismo che permea nei dialoghi del gioco, che come dicevamo, colpisce duramente Metal Gear e prende in giro diversi film d'azione. Un po' come quel capolavoro di UnMetal (recensito nel numero 33 di RMW).

Tirando le somme direi che Metal Dragon è un buon gioco d'azione che rende giustizia all'età d'oro del Megadrive. E' una discreta sfida per i nostri riflessi e ha una sezione tecnica molto meritoria, soprattutto se teniamo conto del gruppetto molto ristretto che ha "partorito" il titolo.

Avrei gradito un po' più di lunghezza e una difficoltà maggiore (solo il livello hardcore è difficile).

di **Carlo N. Del Mar Pirazzini**



## GIUDIZIO FINALE



### » Giocabilità 80%

E' un notevole esercizio di nostalgia, umorismo e passione per i classici "guerriglieri" del passato. Manca di alcuni aspetti, ma il "viaggio" è molto piacevole. L'azione e la potenza di fuoco ci sono in Metal Dragon.

### » Longevità 70%

Ci sono alcune lacune nella curva di difficoltà. I livelli di difficoltà non sono tanto diversificati e i continue infinite/sistema di password ve lo faranno terminare anche troppo presto. I due finali però meritano di essere visti.





# STAR OCEAN

Il Super Nintendo è noto per sfoggiare una libreria piuttosto ampia ed eccezionale di giochi di ruolo, ma alcuni di questi piccoli capolavori non sono mai apparsi al di fuori del Giappone.

Star Ocean è in cima alla lista dei titoli sconosciuti, ma grazie ad un gruppo di "traduttori", è stata creata una patch che permette di giocare il titolo in inglese e/o in spagnolo.

La più grande distinzione di Star Ocean deriva dal fatto che è il culmine del lavoro di diversi sviluppatori che originariamente avevano lavorato al successo di Tales of Phantasia, il che ci spiega perché il gioco sfoggia così tante caratteristiche e stili simili.

Detto questo, vale la pena sottolineare che Star Ocean è un'esperienza di gioco di ruolo distintiva che presenta un sistema di battaglie davvero unico e alcune delle tecnologie più all'avanguardia disponibili per la console a 16 bit di Nintendo al momento della sua uscita. È anche il motivo per cui il gioco tradotto ha impiegato 26 anni per uscire e per risultare compatibile su un emulatore. Ma vi assicuro che tutto questo tempo passato ad aspettare verrà ripagato. Visivamente siamo di fronte ad uno dei giochi per Super Nintendo più belli mai creati. Il gioco originale utilizzava un chip grafico speciale all'interno della cartuccia chiamato S-DD1 che permetteva agli sviluppatori di utilizzare set di grafica complicati

durante il gioco pur adattando il tutto all'interno dei limiti di dimensione della ROM.

Questo è stato uno dei motivi per cui Star Ocean era così difficile da emulare e per molto tempo ha richiesto di numerosi add-on grafici per essere giocato.

Ogni area di gioco presenta un ambiente dettagliatissimo e lussureggiante, un livello di dettaglio che raramente si è visto su un titolo 16 bit. Ogni area è fortemente caratterizzata e ha il suo aspetto e un'atmosfera unici e raramente vedremo ripetuto qualcosa due volte, il che ha l'effetto di rendere il gioco incredibilmente vasto. È chiaro che gli sviluppatori hanno dedicato parecchio tempo ed energia per dare vita ad una grafica del genere e non hanno badato a spese nel farlo. Senza ombre di dubbio Star Ocean è uno dei titoli più belli graficamente per questa console e forse l'esempio perfetto delle capacità visive.

Come se la grafica non bastasse, anche la colonna sonora non smette di stupire. La musica ricorda l'epicità dei titoli di Final Fantasy, ma con un tocco più melodico che lo distingue dalle altre produzioni Enix di quel periodo. Anche in questo caso la varietà musicale è impressionante e nessuna traccia musicale verrà ripetuta all'interno del titolo. Una particolarità del sonoro è l'uso dei campioni audio. Durante la battaglia sentiremo spesso

**Anno:** 1996

**Editore:** Enix/Tri-Ace

**Genere:** Action/GDR

**Piattaforma:** Super Nintendo





urlare e imbastire dialoghi. Per gli standard dei giochi moderni non sembra nulla di spettacolare, ma all'epoca non c'era nulla di simile prima della Playstation.

Ma il punto di forza di questo gioco è il sistema di combattimento in tempo reale davvero unico che mescola il sistema di combattimento d'azione ad un motore di combattimento a turni più tradizionale.

Si possono eseguire attacchi in tempo reale, ma invece che brandire una spada, potremo selezionare un comando d'attacco. Potremo prendere di mira i nemici specifici, nonché utilizzare oggetti e attacchi magici speciali che possono essere assegnati ai pulsanti dorsali del nostro pad. Si controlla un personaggio alla volta, ma potremo scorrere i personaggi disponibili e cambiare quello che si sta utilizzando in qualsiasi momento della battaglia. Gli altri PG saranno controllati dalla IA del gioco.

Come nei più classici giochi di ruolo, anche in Star Ocean si trascorrerà molto tempo ad esplorare le aree di gioco e, occasionalmente, lanciarsi in qualche quest secondaria per acquisire esperienza. Queste missioni aggiuntive che il gioco chiama "Azioni private", non sono determinanti al completamento della storia, ma aggiungono moltissimi dettagli e profondità alla trama,

rendendo il titolo meno lineare.

È Facile capire perché così tanti fan dei giochi di ruolo su Super Nintendo hanno tenuto questo titolo in così grande considerazione per così tanti anni.

Facile anche capire perché è stato uno dei primi giochi ad essere tradotto in inglese dal team DeJap. Ora che la grafica del gioco è stata sistemata e decompressa, non c'è alcun motivo per cui i giocatori non dovrebbero patchare la rom e provare questo gioiello.

Star Ocean è facilmente uno dei migliori giochi di ruolo disponibili per la console a 16 bit e si posiziona sul gradino più alto assieme a The Legend of Zelda: A Link to the Past.

Negli anni è uscita anche una bella versione per PSP che però risulta meno affascinosa di questa versione.

Provatelo.

di **Takahiro Yoshioka** e **Carlo N. Del Mar Pirazzini**

## GIUDIZIO FINALE



### » Giocabilità 98%

Sistema di controllo incredibile e un mix tra azione pura e combattimento a turni. Letale ma incredibile.

### » Longevità 95%

Un titolo longevo e vastissimo da esplorare e da scoprire. Bellissime le quest di contorno.





**NEW GAME**

# ATTACK OF THE PETSCII ROBOTS

**Anno:** 2021/2022  
**Editore:** The 8 Bit Guy  
**Genere:** Action  
**Piattaforma:** C64/VIC 20/PET/Amiga/APPLE II/PSP  
**Recensione:** Commodore 64  
**Sito Web:** <https://www.the8bitguy.com/>

I robot hanno preso il controllo degli insediamenti umani su un certo numero di pianeti e spetta a noi, impavidi eroi, infiltrarci e distruggere tutte le macchine ribelli.

Per fare tutto questo, tuttavia, sarà necessario un po' di strategia e il nostro compito sarà di esplorare l'ambiente di gioco e cercare di individuare le armi, i kit sanitari e le card per accedere ad alcune zone.

Ogni robot presenta un livello di sfida differente: alcuni sono molto deboli e possono essere eliminati con l'arma base altri sono più ostici e astuti e, per farli fuori, dovremo agire in modo scaltro ed utilizzando armi più potenti.

Sono disponibili 9 mappe di gioco che richiedono grande tattica e studio e un ottimo grado di sfida. Attenzione però l'esplorazione è "letale". Non è presente un sistema di salvataggio o password e, se arriviamo alla morte, ci toccherà ricominciare da capo e ripartire dall'inizio.

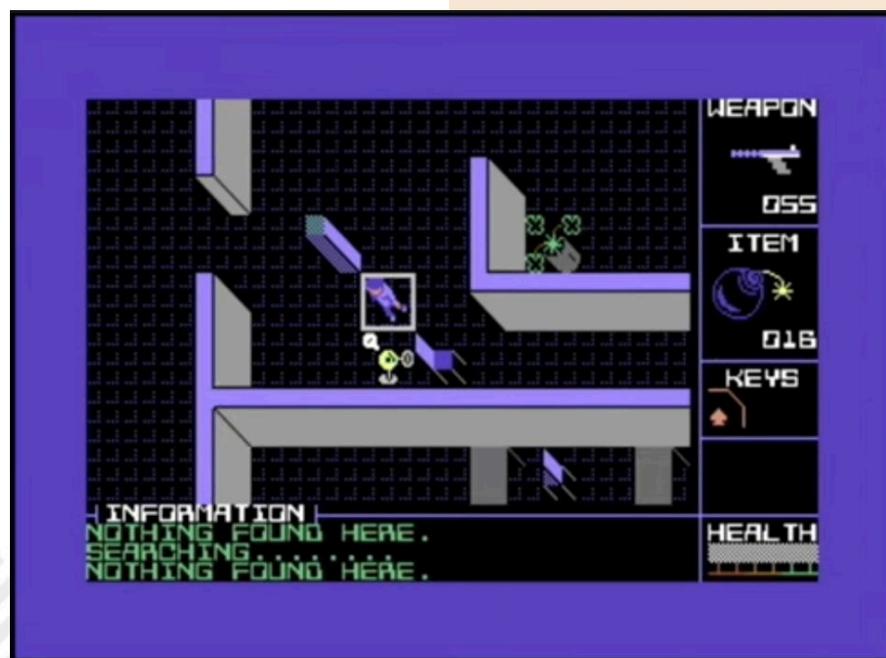
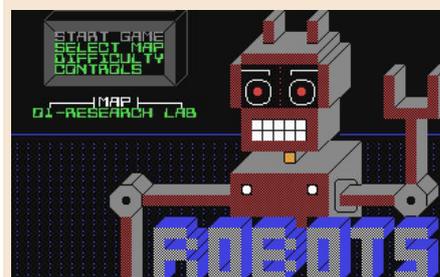
Attack of the PETSCII Robots ha un aspetto e una sensazione unica rispetto ai giochi attuali. Simile a quella che trovavamo nei titoli di metà degli anni '80.. un fascino nostalgico.

La sua grafica fa un lavoro egregio (è possibile giocare il titolo in tre modalità grafiche classica, PET o Color



PETSCII). Ben animata e indubbiamente d'effetto per questo titolo. Lo scorrimento è un po' a scatti, tuttavia con un campo visivo limitato c'è un po' di uniformità e non infastidisce più di tanto.

Avere una maggiore varietà tra i robot





## GIUDIZIO FINALE

### » Giocabilità 80%

Un sacco di elementi da cercare, 9 livelli e una serie di modi interessanti e diversi per eliminare i robot.

### » Longevità 60%

Solo tre varietà di nemici presenti e la mancanza di un sistema di salvataggio abbattano pesantemente la longevità finale. Peccato.

avrebbe migliorato maggiormente il gioco aumentando la sfida.

Ci sono diversi modi di far fuori i robot e cercare oggetti durante l'avventura risulta divertente.

Il vero problema è la mancanza di un sistema di salvataggio o un checkpoint o una password. Un danno per il gioco, perché ogni livello si gioca esattamente

allo stesso modo ogni volta e quando si muore ci viene chiesto di ripetere esattamente lo stesso identico processo precedente.

Questo deficit abbassa molto la longevità finale.

Rimane comunque un buon titolo.

di **Roberto Del Mar Pirazzini**





# KLONOA 2 LUNATEA'S VEIL

Anno: 2001

Editore: Namco

Genere: Platform

Piattaforma: Playstation 2

È bello andare a casa di Nith, nel caos della sua soffitta, tra computer smontati, console da gioco, non so quanti Commodore 64 (ne ho solo 6! NdN) ogni tanto ci si perde.

Nel mio periodo di "reclusione" da Covid, ho passato un po' di tempo a guardare nel mio tempo libero da studentessa qualche titolo interessante ed ecco, che tra diversi titoli e copertine, scorgo questo Klonoa 2. Perché ne sono attirata? Mi viene in mente che da piccolina lo giocavo nella Playstation di mio padre, ma questo è il seguito e volevo provarlo. Accendo la Play 2 di Nith e parte la magia.

Il primo Klonoa mi aveva proprio preso. Sarà per il protagonista simpatico e buffo e per quel suo mondo



a cartoni animati, ma mi ci ero proprio applicata fino a finirlo.

La storia di Klonoa 2 segue il protagonista dalle orecchie flosce attraverso le sue avventure nel mondo onirico di Lunatea. Questo mondo incantato si mantiene in equilibrio attraverso le quattro campane dell'armonia, situate nei quattro diversi regni. Ma una terribile forza malvagia





## GIUDIZIO FINALE

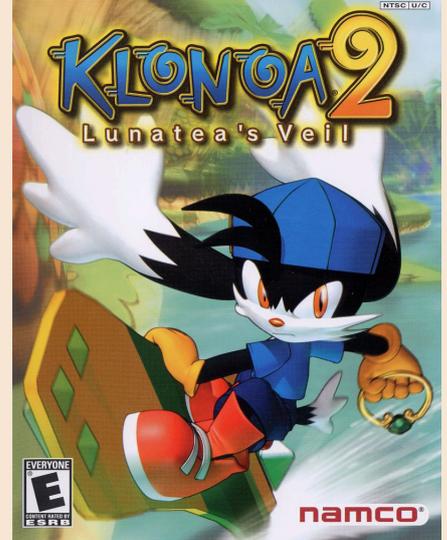
### » Giocabilità 80%

Divertente la meccanica ripresa dal primo capitolo per Play1 e le nuove aggiunte.

### » Longevità 40%

Tarato per un pubblico davvero giovane. Si finisce e ci lascia disperate e con la voglia di un altro pezzo di gioco.

PlayStation 2 



sta progettando di portare una quinta campana, quella del Dolore, che farà ammalare le sacerdotesse e apparire numerosi mostri in tutto il reame. Spetta a Klonoa e ai suoi due nuovi compagni – Lolo una giovane maghetta e Popka uno strano personaggio con gli occhi di vetro – impedire che questo complotto malvagio si realizzi.

Ok la storia sdolcinata da New Age giapponese l'ho riportata, ora parliamo del gioco in sé.

Non è un titolo originale. E' un gioco di piattaforme che riprende quello visto nel primo capitolo aggiungendo un paio di dinamiche nuove. Come nel primo, potremo afferrare i nemici stile Yoshi usando il nostro anello del potere e usarli per spingerci più in alto rispetto al normale salto, oppure usarli come proiettili da lanciare contro altri cattivi o per attivare interruttori o blocchi in mezzo al cammino. Ci sono diversi nuovi nemici in Klonoa 2 che ci garantiscono abilità speciali una volta catturati, come il Kiton che funge da elicottero per un breve periodo, o il boomie che diventa un esplosivo.

Il gioco mantiene sempre un buon equilibrio tra i semplici enigmi e i salti da una piattaforma all'altra. Insomma è sempre divertente.

Peccato per la sua devastante mancanza di difficoltà. Il gioco sembra calibrato

per bambini molto piccoli e in meno di 6 ore lo si porterà a termine. Peccato davvero perché il ritmo di gioco è piacevole e vorresti che durasse almeno il doppio.

Bella la grafica, colorata e veloce. Bello il sonoro... Semplici i comandi. Ma porca miseria!! Questa mancanza di difficoltà mi ha lasciata basita.

Peccato davvero. Un buon platform ma nulla di più.

di **Ingrid Poggiali**





# TWINKLE TALE

Anno: 1992

Editore: Wonder Amusement

Studio – ZAP Corporation

Genere: Run and Gun

Piattaforma: Sega Megadrive

Retromagazine World, il piacere della scoperta. Continua il nostro viaggio tra i titoli sconosciuti per alcune piattaforme. Dopo aver visionato alcuni shoot em up per NES oggi è la volta del Sega Megadrive.

Twinkle Tale è un run and gun con visione dall'alto rilasciato nel 1992 esclusivamente sul mercato giapponese.

Si distingue dagli altri titoli del suo genere per la sua ambientazione unica: combina un gameplay che ricorda Mercs o Knightmare di Konami (avete letto la recensione sullo scorso numero della versione Amiga? NdN) con un simpatico tema horror-kawai che ci ricorda la serie Cotton.

Affronteremo ogni livello falciando ogni cosa in vista e assorbendo ogni power up possibile con il nostro personaggio, la simpatica stregghetta di nome Saria, completa di un bel bastone magico e cappello a punta.

I nemici sono un mix tra i mostri fantasy classici e quelli della mitologia nipponica e ogni livello ne presenta di caratteristici e soprattutto, a flotte!!! La schermata del titolo mostra un'isola galleggiante e la schermata introduttiva che spiega la storia, realizzata per somigliare ad un libro illustrato, cosa che dona un perfetto tono al gioco.

Prima di tutto, Twinkle Tale è un gioco rapido e veloce. Saria si trova fin da subito attaccata da una quantità di nemici feroci e velocissimi. Per sua



fortuna (e nostra) è dotata di un'arma standard solitaria e di una serie di armi secondarie che vengono rilasciate dai nemici affrontati. Il pulsante C del nostro pad passerà da un'arma all'altra e il pulsante B farà fuoco con l'arma attualmente selezionata.

La Shooting Star spara un ampio flusso ad alta potenza direttamente





di fronte a noi, la Diamond Arrow spara in più direzioni e contemporaneamente e la Silver Comet è un tiro di fuoco leggermente inferiore ma ha capacità di tracciamento del nemico. Questo rende il titolo molto vario e dona una discreta varietà rispetto ad altri titoli simili. Non c'è un'arma migliore dell'altra, ma scopriremo che generalmente otterremo risultati migliori con armi specifiche contro determinati nemici. Questa meccanica è stata successivamente utilizzata per alcuni livelli dei titoli come Radiant Silvergun e Neo Contra.

Ogni arma può essere potenziata trovando delle stelle nascoste dentro ad alcuni scrigni del tesoro, che aumenteranno la potenza di fuoco fino a tre volte. Se si viene colpiti, prima di morire e andare in game over, la barra d'energia si abbasserà grazie al numero di potenziamenti che otterremo con l'arma.

Inoltre sono presenti le essenziali bombe, e sono di due tipi: la prima ha una capacità di ricerca mentre la seconda crea un Krakatoa a schermo intero. Entrambe si innescano con il tasto A.

Ci sono nove livelli in totale, tutti piuttosto impegnativi. È possibile modificare la difficoltà iniziale nelle opzioni di gioco ed è possibile anche attivare un piccolo trucco che ci permette di saltare al livello desiderato (nella schermata iniziale del libro di fiabe che vediamo dopo aver premuto start, premiamo Su, Sinistra, A e C e poi start e potremo selezionare il livello col D-pad).

I combattimenti contro i boss sono splendidi. Il boss di ogni sezione viene affrontato come in un classico sparatutto verticale, con il personaggio in grade di sparare solo verso l'alto. Ogni boss è enorme, con schemi di movimento e schemi di fuoco che riportano alla

mente ciò che ci si aspetta da uno shooter. Se siete fan di questo genere è una vera soddisfazione.

Un'altra cosa che spicca davvero in Twinkle Tale è il livello di varietà nelle sue fasi, che implica costantemente più del semplice movimento nella direzione appropriata e far saltare in aria le cose. Un esempio particolarmente valido di questo fatto è il terzo livello, dove si dovrà attraversare una serie di passerelle di pietra nel cielo. A differenza della maggior parte dei giochi simili, in cui si può camminare lungo il bordo senza preoccuparci troppo di cadere, qui se non si fa attenzione si precipita verso il game over. Occorre tempismo, concentrazione e manualità.

Il gioco è stato realizzato negli anni centrali della vita del Megadrive, ma sembra un titolo per la generazione di giochi successiva per questa console. La grafica, sebbene ancora granulosa rispetto ad un titolo Snes, è tra le migliori su questa piattaforma. Coloratissima, con sprite animati e ben dettagliati e con degli sfondi progettati in modo fantastico. Molti livelli mostrano scorci del mondo e sono incredibili.

Il design del Boss è bellissimo. Concepiti in modo eccezionale e animati in modo impeccabile.

Carino il supporto audio ma non incisivo, alla fine blastare nemici non richiede chissà che tipo di musica d'accompagnamento.

È inspiegabile come questo gioco sia quasi sconosciuto fuori dal Giappone e quanto poco successo ha ottenuto. Siamo di fronte ad una perla per questa console che consiglio a tutti di provare. Sulla rete è possibile trovare anche una versione patchata in inglese, non è determinante ma se volete comprendere i dialoghi può essere utile.

di **Roberto Del Mar Pirazzini**

## GIUDIZIO FINALE

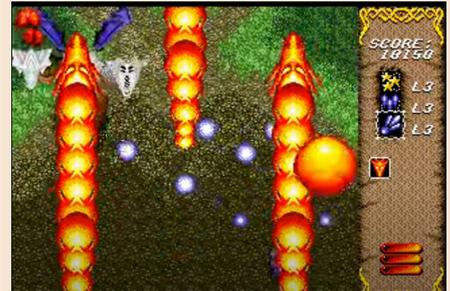


### » Giocabilità 90%

Siamo di fronte ad una perla per questa console che consiglio a tutti di provare.

### » Longevità 80%

Spicca davvero in Twinkle Tale il livello di varietà nelle sue fasi.





# ZAKU

**Anno:** 2009

**Editore/Sviluppatore:** Super Fighet Team - PenguinNet

**Genere:** Shoot'em up

**Piattaforma:** Atari Lynx

Zaku è uno di quei giochi che se fosse uscito nei tempi d'oro di questa console sarebbe balzato in testa alle classifiche di apprezzamento e vendita, forse rilanciando il portatile sul mercato dominato dalle rivali Nintendo e Sega. Ma Zaku è un gioco nuovo, uscito solo nel 2009, circa un decennio e mezzo dopo l'uscita di scena del Lynx dal mercato.

Sviluppato da Osman Celimli di PenguinNet e pubblicato da Super Fighter Team, è uno shooter a scorrimento orizzontale che ricorda il bellissimo Air Zonk per PC Engine. Nonostante la sua natura da homebrew, ha avuto un lancio commerciale davvero importante, partendo proprio da un vero e proprio kit di sviluppo specifico per il portatile Atari.

Lo hanno messo in vendita in una cartuccia e confezionato ad arte con tanto di manuale di istruzione ben fatto. Il gioco è un vero e proprio omaggio ai primi shooter degli anni Novanta. Il personaggio è caratterizzato proprio come le varie mascotte di quel periodo e ricorda nell'aspetto lo stile irriverente di Sonic e Bonk.

La grafica è coloratissima e ben realizzata e c'è uno scrolling in parallasse davvero eccelso. Davvero un lavoro ammirevole.

Quello che amo di questo gioco è la sua enorme quantità di battaglie con



i boss. Amo combattere questi personaggi enormi ed assurdi. Ci sono almeno tre boss per fase, con il livello finale composto interamente da combattimenti con più boss consecutivi. Certo alcuni di loro sono simili, ma c'è comunque tantissima varietà da sembrare sempre originali. Alcuni dei miei cattivi preferiti sono ad esempio il pesce colletto blue, il





## GIUDIZIO FINALE

### » Giocabilità 90%

Controlli perfetti, ottimo gameplay e battle boss bellissime!

### » Longevità 85%

Forse corto rispetto le aspettative ma la difficoltà si bilancia tra livelli semplicissimi e battaglie coi boss davvero impegnative.



tostapane volante spara crostini e un programma per computer che produce continuamente e in massa altri mini boss. Geniale!

Trovo che sia dotato anche di una trama folle ma piacevole. È una storia divertente che narra di uno sviluppatore di giochi malvagio che ha messo le mani su un software prototipo e lo sta utilizzando per inondare il mercato con giochi di qualità discutibile. Spetta quindi a noi nei panni di Zaku recuperare il software e tutte le sue copie. Una trama assurda dal finale davvero ben fatto e divertentissimo che vale la pena di essere visto.

Zaku è un gioco pulito, forse troppo pulito. Non ci sono veri power up, né c'è troppa azione su schermo. I nemici spesso volano verso di noi non più di due alla volta e i proiettili non sono così frequenti come in altri titoli di questo genere.

Anche se questo si adatta al piccolo schermo sfocato del Lynx, significa anche che il gioco perde un po' di quella frenesia tipica proprio di questo genere di giochi

Inoltre alcuni livelli sono di una semplicità assurda mentre certi boss vi faranno elencare le maggiori divinità planetarie tanto sono difficili.

Il gioco si termina in circa due ore,

può sembrare breve ma vi ricordo che l'autonomia delle batterie della console si aggira tra le tre e le sei ore di attività e che non è previsto sistema di salvataggio o password. Non credo valga la pena di prolungare la durata complessiva del titolo.

Ad ogni modo la longevità è assicurata dal grado di sfida. Non vi preoccupate. I giochi homebrew come Zaku sono una manna dal cielo per questa piccola console. È un titolo speciale e, probabilmente, il miglior sparatutto mai giocato su Lynx.

Vale la pena cercarlo e giocarci.

di **Roberto Del Mar Pirazzini**





**NEW GAME**

# S.O.L.O.

Anno: 2021

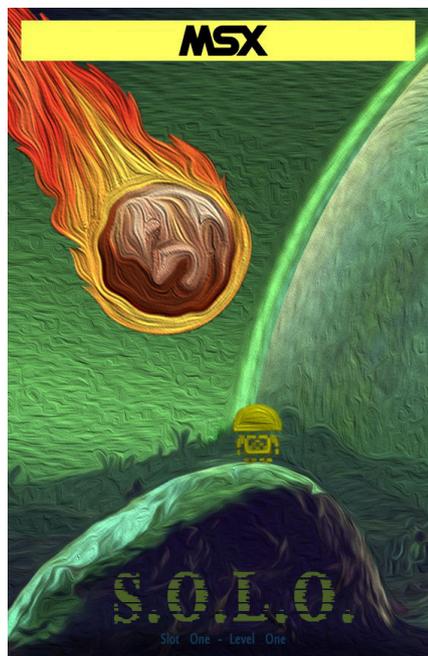
Editore: Rolando

Genere: Platform/Puzzle

Piattaforma: MSX

Sito web: [https://](https://www.msxdev.org/2021/07/16/msxdev21-19-solo/)

[www.msxdev.org/2021/07/16/msxdev21-19-solo/](https://www.msxdev.org/2021/07/16/msxdev21-19-solo/)



Un enorme asteroide è in rotta di collisione con il pianeta. I droidi chiamati S.O.L.O (slot one, level one), nati per l'estrazione di un prezioso minerale chiamato Ether, sono gli unici a poter evitare la catastrofe. Per fare ciò uno di questi robot, il protagonista, dovrà farsi strada tra numerosi livelli pieni di puzzle e pericoli e salvare sé stesso e il mondo intero. Ci troviamo di fronte ad un ottimo titolo nato per l'MSXdev dello scorso anno e che merita di essere giocato. Ha un'atmosfera tipica di questo genere di platform con punte puzzle.



I livelli sono ben strutturati e richiedono una buona dimestichezza e destrezza. Il nostro robot necessita di potenziamenti per poter raggiungere l'uscita di ogni livello.



Semplice da giocare e mai punitivo, S.O.L.O è anche un piccolo gioiellino grafico e con una buona parte audio. Esiste anche una versione per ZX Spectrum anche in questo caso ben fatta.



Devo ammettere di essere rimasto "agganciato" a questo gioco per diverse ore tanto da portarlo a termine con soddisfazione. Non è un gioco semplice ma ti incolla al joystick. Consigliatissimo.

di **Giampaolo Moraschi**



## GIUDIZIO FINALE

### » Giocabilità 90%

Semplice e lineare. I livelli sono tutti ben realizzati e mai senza senso.

### » Longevità 80%

Difficile se lo si sottovaluta. Ma vi terrà inchiodati al tavolo per vedere la fine.





**NEW GAME**

# ROGUE 64



Disponibile da Bitmap Soft e sviluppato da Badgerpunch, Rogue64 è una moderna avventura roguelike che vale davvero la pena provare.

Il gioco utilizza la generazione procedurale dei livelli e ogni partita è un'esperienza unica. Mostri più complessi e sfide più difficili vengono presentate man mano che il gioco procede, assicurando che questa sia un'esperienza di gioco memorabile sul nostro Commodore 64.

Saremo Zendar l'esploratore, alla ricerca di fortuna e gloria nei sotterranei di Mordecoom. Si narra che in questo dungeon sia nascosto un potente oggetto magico e, come tutti gli esploratori, lo desideriamo! L'unico problema sono gli esseri malvagi che abitano nelle caverne in agguato nell'oscurità, in attesa di attaccare ogni visitatore ostile.

Il gioco si basa su Rogue4k, un gioco simile nato per il concorso benefico The C64 "Cassette 50" e che occupava poco meno di 4000 byte.

Il titolo presenta un aspetto grafico minimalista ma funzionale, corredato da un ottimo comparto sonoro di Sami

Louko.

Il joystick in porta 2 controlla il movimento del protagonista, per afferrare il bottino basta camminarci sopra mentre per combattere contro i nemici basta camminare verso di loro. Attenzione... detto in questo modo sembra una passeggiata, ma vi assicuro che il livello di mortalità causato da scelte errate è altissimo. Questo però non abbassa la curva di interesse per il titolo che si dimostra uno dei più validi del genere sul C64. Potete acquistare il titolo in download digitale sul sito oppure potete comprare la scatola Deluxe per 35 sterline sul sito web dei Bitmap Soft. La scatola include la cartuccia, il manuale e una serie di gadget.

Insomma, un bel titolo divertente per il Commodore 64.

di **Carlo Nithaiah Del Mar Pirazzini**



Anno: 2022

Editore: Badgerpunch Games

Genere: Roguelike games

Piattaforma: Commodore 64

Sito web: [https://](https://badgerpunch.itch.io/rogue64)

[badgerpunch.itch.io/rogue64](https://badgerpunch.itch.io/rogue64)



## GIUDIZIO FINALE



### » Giocabilità 90%

Essenziale ma coinvolgente il sistema di controllo. I livelli generati in modo procedurale aumentano la voglia di giocare.

### » Longevità 80%

Difficile ma piacevole. Dopo il game over rischierete il tasto di fuoco più e più volte.





# SIGIL

**Anno:** 2019

**Editore:** Romero Games

**Genere:** FPS

**Piattaforma:** PC

**Sito Web:** <https://romero.com/sigil>

Immaginate se un giorno uno dei 12 apostoli venisse a casa vostra e vi desse un pezzo in più della Bibbia.

Questo è essenzialmente ciò che Sigil è per i fan degli FPS.

È la manna del paradiso di questo genere di giochi, anche se, dato il contesto, l'inferno potrebbe essere un punto di partenza più appropriato per questa metafora.

Sigil è un quinto episodio "non ufficiale" di Doom (o Ultimate Doom), anche se è tanto ufficiale quanto non ufficiale, creato com'è dall'unico DOOMDAD ovvero John Romero stesso.

Questo MegaWad aggiunge al gioco nove nuove mappe per giocatore singolo e nove per il multiplayer, tutte realizzate dalla fervida immaginazione di JR.

Si può scaricare gratuitamente, ma volendo per solo 7 sterline è possibile acquistare la versione speciale che include una colonna sonora personalizzata.

Per tutti quelli interessati agli FPS vecchia scuola, Sigil è un prodotto interessante, anzi eccitante.

Devo ammettere una certa trepidazione quando ho sentito per la prima volta parlare di Sigil.

Felice del coinvolgimento di John Romero, ma allo stesso tempo terrorizzato. Vedete solo perché qualcuno vi sta porgendo un nuovo tassello della Bibbia, o un nuovo seguito di una serie tv o persino una nuova trilogia significa che quel qualcosa sia buono. Prendiamo ad esempio "San" George Lucas, ci ha

donato nuovi frammenti della sua "Bibbia spaziale" e guardate come è andata a finire!

Spesso gli artisti, come tutte le persone, cambiano nel tempo e il loro occhio creativo cambia spesso con esso, certe volte in un modo che lo mette in contrasto con il creatore. Questo non è necessariamente un tratto negativo, ma rende difficile ricreare l'atmosfera iniziale, la perfezione della prima creazione... come ad esempio hanno dimostrato appunto i film di Star Wars o la serie di Alien.

Aggiungiamo a questo che J. Romero non è stato esattamente il creatore più prolifico negli ultimi anni e che Doom avesse bisogno di un episodio in più come se avessimo bisogno di un altro film su Terminator, e Sigil sembrava appoggiare su basi traballanti.

Fortunatamente il gioco è eccellente. Non solo è bello e soddisfacente, ma estremamente divertente e con una sequenza di mappe stimolanti,





inquietanti (a volte davvero inquietanti!) e giocabilissime.

Ci ricorda ciò che rese Doom nel 1993 eccezionale.

Sigil ci butta nella mischia già dai primi secondi, gettandoci in una stanza rossa con un pentagramma scavato nel pavimento, aggrediti da tre Imp. Subito veniamo messi sotto pressione, sentendoci sull'orlo della morte, un tema che continua in tutto Sigil.

Sigil è la forma originale di Doom. Scordatevi i survival fps o i seguiti pompanti dalla grafica. È Doom in una sua forma più infernale e ostile.

È un terrificante massacro METAL!! È il regno distopico-demoniaco partorito dalla mente di John Romero, fatto di viscere e sangue, pareti con le facce e intere stanze che si contorcono.

Vale la pena notare che Sigil è ricco di intensi scontri a fuoco. Ogni stanza è piena di nemici da far saltare in aria. Ma non si limita a schiantarci contro intere legioni di nemici, schiera con cura combinazioni specifiche di avversari che ci faranno pensare.

Dovremo agire pensando al modo

migliore per uscire fuori da questo inferno.

La mia principale critica al gioco è che non aggiunge nuove armi o nemici. Progettato come una celebrazione del Doom originale e di tutte le parti in esso contenute, il che è abbastanza giusto. Tuttavia, sarebbe stato bello vedere qualcosa di nuovo, solo per rendere le cose più piccanti.

Infine, e questo dovrebbe essere ovvio, ma se non avete mai giocato a Doom prima, dovrete farlo prima di avventurarvi in Sigil.

È un gioco per veterani di Doom, con un grado di sfida alto e che forse i nuovi giocatori potrebbero trovarlo "travolgente".

Nel complesso, il gioco mi è piaciuto molto di più di quanto mi aspettassi. Qualcosa di più di un semplice tributo a un classico; mescola e accresce tutti gli elementi che hanno reso grande l'originale, risultando in un raro esempio di follow-up a lungo ritardato che dà un contributo significativo all'opera originale.

di **Carlo N. Del Mar Pirazzini**

## GIUDIZIO FINALE



### » Giocabilità 99%

Nuove mappe stesso incredibile e frenetico gameplay!!

### » Longevità 90%

Difficile, crudele, ostico e straniante. Non è perfetto ma è un capolavoro.





NEW GAME

# WITCH N' WIZ

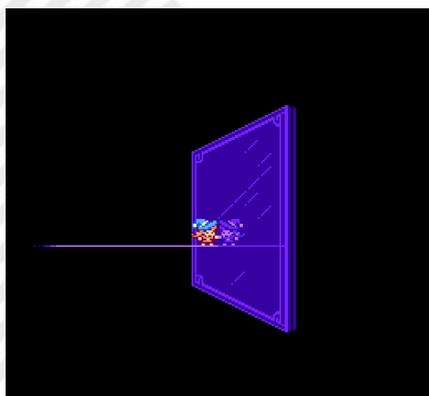


Ma quant'è simpatico questo gioco!

Witch 'N' Wiz presenta una giovane stregghetta che è fuori a salvare la sua amica, rapita dalla malvagia Strega dell'oscurità.

Dopo l'introduzione "giocabile", ogni capitolo è diviso in più livelli.

La nostra paffuta strega può arrampicarsi e spingere dei blocchi soggetti a gravità. Il loro corretto posizionamento determina la riuscita del livello e il passaggio a quello successivo!



Se si commette un errore è possibile riavvolgere il livello con il tasto B e riprovare.

Il grado di sfida degli enigmi per ogni singolo quadro di gioco è ben strutturato. Alcune volte ci si metterà davvero poco a passare alla fase successiva, altre volte invece dovremo spremere forte le meningi per cercare di risolverli, però non sono mai impossibili.

Carina la grafica degli sfondi e i personaggi molto "KAWAI".



La versione in formato su cartuccia è ben curata con un manuale colorato e un ottimo confezionamento.

Mi sono divertita.

di Ingrid Poggiali

Anno: 2021/2022

Editore: Matt Hughson

Genere: Puzzle

Piattaforma: Nintendo NES

Sito web: [bit.ly/witchnwiz](https://bit.ly/witchnwiz)



## GIUDIZIO FINALE

### » Giocabilità 87%

Sistema semplice di gioco e divertente. Non presenta difficoltà di apprendimento.

### » Longevità 87%

Un bel grado di sfida! Vi farà divertire per molto tempo.





# DORAEMON

Ok ok, io ho un vero problema con Doraemon!



Cioè a me il gatto spaziale mi piace un sacco e questo è uno di quei pochi giochi kart da avere in collezione.

Tra l'altro il manga di Fujiko F. Fujio è del 1969! Lo sapevate? Sicuramente i super nerd si (puoi starne certa dada, NdNith)! Poi tutti in fondo abbiamo visto l'anime in tv. Ma passiamo al gioco!



Come ogni gioco kart Nintendo, puoi prendere degli oggetti da usare durante la gara, ma non aspettatevi grandi grafiche, è curata ma è pur sempre un piccolo Game Boy!

Si possono scegliere i vari personaggi: Doraemon, Nobita Nobi, Shizuka

Minamoto, Takeshi Goda e Suneo Honekawa, Dorami è sbloccabile.

Si può giocare in multiplayer e questo è divertente.

Forse non è lunghissimo né difficile, però diverte.

Su Game Boy Color esiste il 2 se non sopportate i toni verdognoli/grigi che offre il GB.

E voi avete mai giocato a questo?



di Barbara "Morgana" Murgida

Anno: 1998

Editore: Epoch

Genere: Driving game

Piattaforma: Game Boy



そのの のび太は ひとぼんじゅう  
あたらしいコースを  
かんがえていました。



## GIUDIZIO FINALE



### » Giocabilità 80%

Semplice e piacevole da giocare.

### » Longevità 65%

Non ci vuole tanto tempo per vedere la schermata finale dei titoli, però è presente il multiplayer che allunga la vita del gioco (giocabile con il cavo se avete due Gb oppure con Kaillera in emulazione, NdNith).





# SKI

Anno: 1983

Sviluppatore: Commodore

Genere: Simulazione sportiva

Piattaforma: Commodore 64

La primavera è alle porte e si sente aria di rinascita, non a caso la nazionale italiana di rugby ha vinto la partita contro il Galles al Sei Nazioni dopo una valanga di sconfitte consecutive; che sia un buon segno anche per questo sottovalutato sport nel nostro paese?

Per essere il primo giorno di primavera, a Milano fa ancora freddo a meno di non trovarsi nel primo pomeriggio sotto al Duomo. Grazie a questo freddo un po' anomalo, (o forse l'ultimo della stagione è quello più forte?) mi sono ricordato di un altro sport poco seguito se non ai tempi di Alberto Tomba.

Parlo ovviamente dello sci, con le tutte le sue categorie, come lo slalom gigante, lo speciale, la combinata... Esattamente come il qui presente Ski per Commodore 64 anche se, qui di categoria ne esiste solo una e per un gioco del genere più completo bisogna aspettare un bel po' di anni.

Qualsiasi sport vi venga in mente, potete starne certi che sul nostro biscottone qualcosa sia stato realizzato all'epoca, con risultati più o meno soddisfacenti.

A livello globale questo gioco è molto divertente come la musichetta che lo accompagna; sei livelli impegnativi in cui si deve giungere al traguardo entro il tempo stabilito, superando ostacoli come alberi, rocce, lepri che attraversano il percorso, senza dimenticare di passare in mezzo alle bandierine! Non sono molto esperto di sci ma se dovessimo mancarle c'è una penalità sul tempo con conseguente sforamento e squalifica.

Sei livelli di intrattenimento, lavoro manuale e pazienza che terranno occupati voi e i vostri amici e/o

famigliari sfidanti, per interi week end e feste comandate.

Il primo livello diventa abbastanza semplice dopo la seconda partita, mentre i successivi saranno per veri esperti di joystick e tempistiche. Le piste potrebbero indurvi a pensare che sia un gioco ripetitivo, invece sono ricche di particolari che creano piccole differenze. Per esempio la lepre che attraversa il percorso dove pur di salvarle la vita, sacrificherete quasi la vostra contro un albero... Inoltre i tempi di arrivo diventano più corti tra una discesa e l'altra.

Finora, giochi simili li avevo notati solo sui computer moderni o all'interno di giochi olimpici come Winter Games della Epyx. Ciò lo rende particolare, anche se poteva essere sfruttato meglio dal punto di vista competitivo.

Graficamente non mi sento di criticarlo perché per un gioco degli anni '80, le caratteristiche grafiche erano già buone. Dovremo aspettare i gioconi cartonati per vedere qualche miglioria a livello tecnico.

Carino ma non ai massimi livelli il sonoro che serve più che altro a tenere compagnia in un gioco come questo, soprattutto in caso di vittoria e se provate a giocarlo senza musica e con il solo effetto sonoro dello schianto, vi ricrederete presto.

Spero che con questo bel simulatore di sci possiate salutare serenamente l'inverno e soprattutto divertirvi con la varietà di titoli per tutte le macchine che la nostra gloriosa e mondiale rivista sta offrendo da anni.

di **Daniele Brahimi**



Omaggio ad Alberto Tomba

## GIUDIZIO FINALE

» **Giocabilità 75%**  
Divertente fino ad un certo punto. Da soli o con gli amici.

» **Longevità 70%**  
Sei tosti livelli possono invogliare o demotivare. Dipende da voi...





# CRACK DOWN

**Anno:** 1989

**Sviluppatore:** SEGA

**Genere:** Top Down Shooter

**Piattaforma:** Arcade

Protagonista della storia è Mister X, uno scienziato pazzo che ha deciso di creare degli umanoidi per conquistare il mondo. Ecco allora che fanno la loro comparsa Andy e Ben, due agenti speciali che devono infiltrarsi in varie aree prima di raggiungere il laboratorio dello scienziato e distruggerlo.



Il gioco è un top down shooter con la classica visuale dall'alto, in perfetto stile Gauntlet. Presenta un HUD davvero originale, con lo schermo diviso in due parti, ciascuna per ogni player, con tanto di indicatore della vite, delle varie armi disponibili ed una bella mappa al centro che ci indica i vari punti strategici.

Scopo principale del gioco è quello di piazzare diverse bombe all'interno di ogni livello. Innescato l'ultimo ordigno, abbiamo poi un tempo limitato per raggiungere la via di fuga prima dell'esplosione.

Prima di raggiungere il laboratorio di Mister X, dobbiamo attraversare ben quattro mondi, ognuno formato da altrettante aree.

A nostra disposizione abbiamo un fucile, un lanciamissili che possiamo raccogliere lungo il nostro cammino e alcune bombe che distruggono tutto quello che è presente sullo schermo.

Se terminiamo i proiettili, dobbiamo avanzare facendoci strada a suon di calci.

Il gioco è a metà tra uno sparattutto e uno strategico, infatti possiamo ragionare e trovare percorsi alternativi per piazzare tutte le nostre bombe. Certo il timer che scorre non è nostro alleato e dobbiamo essere abili ad applicare la nostra strategia.

Seppur partendo da un'idea già vista, Crack Down presenta una grandissima giocabilità unita anche ad una grafica pixel art davvero ben fatta e a musiche ed effetti sonori che ben si sposano con l'architettura del gioco.

Alla sua uscita in Giappone, le riviste del periodo lo hanno recensito con ottime valutazioni, addirittura azzardandosi a dire che presenta un gameplay che crea dipendenza.

In effetti più ci giochi, più hai voglia di andare avanti nell'avventura per raggiungere il livello finale. Se poi affrontato in modalità cooperativa, il divertimento è ancora più coinvolgente. Crack Down ha ricevuto diverse conversioni per console e computer. Davvero degne di nota quelle per Megadrive e Amiga che, seppur con sfumature grafiche diverse, sono riuscite a mantenere lo stesso originale e appassionante gameplay.

di **Querino Ialongo**



## GIUDIZIO FINALE

### » Giocabilità 90%

Hanno scritto che questo titolo ha un gameplay che crea dipendenza. In effetti la sua giocabilità è incredibile, ancora di più se giocato in cooperativa.

### » Longevità 90%

Come gran parte degli arcade, Crack Down ha un longplay che si aggira sui trenta minuti, ma questo non scalfisce per nulla la sua bellezza. Questo titolo merita di essere riscoperto, nonostante siano passati circa 35 anni dalla sua pubblicazione.



## "SEI TU UN VERO RETROGAMER?"

Come immagino molti dei nostri lettori (siete carinissimi, grazie!), frequento abitualmente diversi gruppi sul noto social blu. Gruppi creati e frequentati da chi, come me, è appassionato di videogiochi, più in particolare di retrogame, retrocomputer, retroconsole e qualunque altra cosa riguardi questo fantastico mondo e che inizi con "retro". Sono spesso un'ottima fonte di notizie, approfondimenti, luoghi virtuali dove discutere, dove qualcuno all'improvviso tira fuori quel gioco che avevi rimosso dalla memoria, dove è possibile trovare informazioni tecniche e curiosità. O semplicemente dove abbandonarsi ai ricordi dei cari bei vecchi tempi, parlando di questo o quel videogioco, magari raccontando di quell'incredibile record stabilito nel lontano 1991.

Tutto molto bello, ma capita, non di rado, di imbattersi in quelli che potrei definire gli "integralisti del retrogaming". Sono coloro che assegnano la patente del vero appassionato, che ovviamente detengono, e che ti dicono come si fa davvero. Sono persone che in realtà si trovano in qualunque ambito, non solo nel mondo del retrogame e, a dirla tutta, non se ne incontrano solo sui social, ma diciamo che da quelle parti è più frequente. Faccio un esempio, giusto l'ultimo a cui ho assistito: finalmente sta per uscire l'A500 Mini (nel momento in cui scrivo, quando leggerete sarà già in commercio), che segue il trend iniziato dal mini NES, proseguito poi dal mini SNES, dal C64 e via via tutte le altre mini console. Ho letto e leggo ancora diverse critiche sul prodotto in sé, a mio parere alcune legittime e altre meno, ma non voglio soffermarmi su questo adesso (ne parleremo abbondantemente nel prossimo numero di RMW, garantito!). Ciò che trovo davvero strana è l'assegnazione del distintivo del vero retrogamer di cui parlavo. Da ciò che ho letto posso dedurre che: se giochi con una di queste mini console non sono un vero appassionato, sono addirittura un gradino sotto chi usa gli emulatori, perché ho speso pure soldi che potevo risparmiare. Interviene poi qualcuno e dice che no, se non giochi su real hardware non sono un vero appassionato. Si inserisce però un altro ancora che ti dice che ok real hardware, ma se non uso i floppy (rimanendo in ambito Amiga) non sono nessuno. E però no, se l'hardware non è esattamente quello che avevo 30 anni fa, non è mica la stessa cosa e probabilmente, con i prezzi che girano oggi, mi sono fatto pure spennare.

La lista potrebbe andare avanti ma mi fermo qui, il concetto credo sia chiaro. Esisterà sempre quello più "puro" di te, che la sa più lunga, apparentemente dotato di radar per trovare il post giusto dove "infilarsi" e pontificare/polemizzare. Ciò che spesso sembra venga dimenticato è che sì, insomma, sono giochi.

E allora giochiamo! PC, console, hardware originale ok ma anche no, emulatori, telefoni, quello che volete, va tutto bene. Anche il forno a microonde se vi pare e sì, qualcuno l'ha fatto davvero. Godiamoci la nostra passione senza pensarci troppo su, facciamoci investire da quella sana botta di nostalgia che una manciata di pixel sa darci. Torniamo per qualche ora gli spensierati ragazzini di un tempo, magari sfidando in doppi infiniti i nostri amici. Facciamo conoscere ai nostri figli e alle nostre figlie la gioia che i giochi "di una volta" ci hanno donato e condividiamola con loro. Evitiamo le inutili polemiche, i litigi, le discussioni infinite che non portano a nulla. Ma se proprio dovessimo finire in mezzo a tutto ciò, e prima o poi capita, allora c'è solo una cosa da fare per risolvere definitivamente la questione e decretare il vincitore: sfidone a Sensible Soccer?

Insomma care lettrici e cari lettori, divertiamoci, ridiamo, abbracciamoci forte e vogliamoci tanto bene. Giochiamo più che possiamo senza preoccuparci di qualunque altra cosa, in fondo è per questo che lo facciamo, no? I nostri amati videogiochi ci offrono la possibilità di evadere da tutto almeno per un po', mettere un attimo da parte la zavorra fatta di mille problemi che tutti, chi più chi meno, ci portiamo dietro quotidianamente, facendoci immergere in mondi colorati e spesso assurdi, in cui l'unica cosa che conta è superare il prossimo livello. Tutto ciò racchiuso in un pugno di bit, davvero niente male, no?

**Beppe Rinella**

## Disclaimer

RetroMagazine World (fanzine aperiodica) è un progetto interamente no profit e fuori da qualsiasi circuito commerciale. Tutto il materiale contenuto è prodotto dai rispettivi autori e pubblicato grazie alla loro autorizzazione.

RetroMagazine World viene concessa al pubblico con licenza: Attribuzione - Non commerciale - Condividi allo stesso modo 4.0 Internazionale (CC BY-NC-SA 4.0 INT) <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.it>

In pratica sei libero di: condividere, riprodurre, distribuire, comunicare o esporre in pubblico rappresentare, eseguire e recitare questo materiale con qualsiasi mezzo e formato, modificare, rielaborare, trasformare il contenuto e basarti su di esso per altre opere, alle seguenti condizioni:

### Attribuzione

Devi riconoscere una menzione di paternità adeguata, fornire un link alla licenza e indicare se sono state effettuate delle modifiche. Puoi farlo in qualsiasi maniera ragionevole possibile, ma non con modalità tali da suggerire che il licenziante avalli te o l'utilizzo del materiale da parte tua.

### NonCommerciale

Non puoi utilizzare il materiale per scopi commerciali.

### StessaLicenza

Se rielabori, trasformi il materiale o ti basi su di esso, devi distribuire i tuoi contributi con la stessa licenza del materiale originario.

Il licenziante non può revocare questi diritti fintanto che tu rispetti i termini della licenza.

### Divieto di restrizioni aggiuntive

Non puoi applicare termini legali o misure tecnologiche che impongano ad altri soggetti dei vincoli giuridici su quanto la licenza consente loro di fare.



**RetroMagazine World**  
Anno 6 - Numero 36 - APRILE 2022

**Direttore Responsabile**

*Francesco Fiorentini*

**Vice Direttore**

*Marco Pistorio*

**Coordinatore Redazione/Editing**

*David La Monaca*

**Responsabili Area Web**

*Giorgio Balestrieri*

