



RetroMagazine

World

future days are back



A PIG QUEST

COMMODORE 64



INTERVISTA A
KAZUHIKO NISHI



HARDWARE
ATARI 5200 SuperSystem

CALC RESULT: Lo Spreadsheet per Commodore 64
PROGRAMMAZIONE: Un esperimento in Assembly per Il C64
Ottimizzazione MSX Basic e supporto Assembly
Grafica... Che passion! Parte IV - Il formato grafico KLA e come interagirvi
Nintendo DS: le (S)protezioni -
... rubriche, interviste, recensioni e molto altro ancora!

C'è ancora molto da fare ed abbiamo bisogno di te!

In questi giorni mi sono ritrovato a parlare con una persona che ha conservato alcuni dei suoi compagni di giochi degli anni '80; precisamente un Commodore 64 ed un Commodore 16. Continuando la nostra conversazione, ho capito come a lui avrebbe fatto piacere condividere l'esperienza di giocare con queste macchine insieme a suo figlio.

Fin qui niente di strano, abbiamo visto decine di foto di appassionati di retro-computing mostrare orgogliosi i propri figli giocare divertiti con i computer e console che hanno allietato la nostra infanzia.

Quello che invece mi ha stupito, è il fatto che non avesse la minima idea che queste macchine siano oggi più vive che mai. Dato il proliferare di siti web, pagine social, riviste dedicate al retro-computing e persino l'interessamento della stampa e delle TV generaliste, mi sarei aspettato che l'argomento non fosse così completamente oscuro. Probabilmente il nostro essere parte del sistema ci fa dare molte cose per scontate, ma così purtroppo non è. Ci sono ancora migliaia di persone che non hanno idea che potrebbero rivivere la loro giovinezza... Ed è un peccato che si perdano questa opportunità.

Evidentemente c'è ancora molto da fare. Dobbiamo riuscire a raggiungere quante più persone possibile. E per farlo abbiamo bisogno del vostro aiuto.

RetroMagazine World ha in progetto da diverso tempo ormai di mettere online tutti gli articoli che sono stati pubblicati sulle pagine della rivista. In questo modo la visibilità del nostro lavoro aumenterebbe e di conseguenza potremmo raggiungere sempre più persone.

Ok, ma noi cosa c'entriamo? Purtroppo il nostro tempo è limitato e quel poco che ci avanza già lo dedichiamo alla realizzazione degli articoli, della rivista e di tutti gli altri progetti di RetroMagazine World. Stiamo quindi cercando qualche volontario che vorrebbe far parte del nostro team aiutandoci alla trasposizione degli articoli sul sito.

Se l'idea di collaborare con noi aiutandoci a diffondere la conoscenza del retro-computing ti affascina, non esitare oltre e contattaci tramite i nostri canali ufficiali!

Non ti promettiamo soldi e fama, ma l'eterna gratitudine di migliaia di appassionati. Forse... :-D

Francesco Fiorentini

SOMMARIO

◇ Nintendo DS - le (s)protezioni	Pag. 3
◇ Atari 5200 SuperSystem	Pag. 8
◇ CALC RESULT - lo Spreadsheet per Commodore 64	Pag. 14
◇ Un esperimento in Assembly per il C64	Pag. 18
◇ Grafica... Che passione! Parte IV - Il formato grafico KLA e come interagirvi	Pag. 26
◇ Ottimizzazione MSX BASIC e supporto Assembly	Pag. 31
◇ Due parole con il genio di Kobe	Pag. 34
◇ ListAmiga - I 10 MIGLIORI PLATFORM SU AMIGA	Pag. 37
◇ Neonnoir (Amiga)	Pag. 43
◇ Lester (C64)	Pag. 44
◇ Life of Mars (MD)	Pag. 45
◇ Bufonada (MSX/2 - Spectrum)	Pag. 46
◇ Mad Stalker Full Metal Forth (MD)	Pag. 48
◇ A Pig Quest (C64)	Pag. 50
◇ Aquabyss (Amiga)	Pag. 52
◇ Maria Renarda's Revenge (Amiga)	Pag. 54
◇ S.A.K. (MSX2)	Pag. 56
◇ Mystic Pillars (NES)	Pag. 57
◇ The Firemen (SNes)	Pag. 58
◇ Minky (Amiga)	Pag. 60
◇ Shadow Warriors Ninja Gaiden (GameBoy)	Pag. 61
◇ Crystalis (NES/GBC)	Pag. 63
◇ James Bond 007 Live and Let Die (C64)	Pag. 64

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

- Dr. Andrea Q.
- Carlo N. Del Mar Pirazzini
- Daniele Brahimi
- Francesco Fiorentini
- Leonardo Miliani
- Roberto Del Mar Pirazzini
- Ingrid Poggiali
- Giampaolo Moraschi
- Giuseppe Rinella
- Eugenio Rapella
- Germán Gómez Herrera
- Takahiro Yoshioka
- Barbara "Morgana" Murgida
- Immagine di copertina: **Giuseppe Mangini**
- Layout di copertina: **Carlo N. Del Mar Pirazzini**





Nintendo DS - le (s)protezioni

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

Canale youtube: <https://www.youtube.com/channel/UCew0CQ8LKya9jVvWXkEwp4Q>

Ed eccoci ad una nuova puntata di questa a quanto pare apprezzata rubrica.

Dopo aver trattato l'oramai non più misterioso iQue Player, oggi cerchiamo di svelare cosa si cela dentro la successiva console portatile di casa Nintendo.

Uscita al termine del 2004 e "potenziata" successivamente con la versione "Lite" nel 2006, il Nintendo DS inizia a presentare diverse caratteristiche che poi ritroveremo nelle future console di questa grande azienda produttrice di videogiochi e consoles.



A destra della foto potete vedere la versione "Fat" mentre a sinistra quella "magra" Lite.

Le principali differenze tra le 2 versioni sono le seguenti. Il DS Lite è:

- circa il 42% più piccolo ed il 21% più leggero pur mantenendo tutte le caratteristiche del DS "normale"
- la sua retroilluminazione è sempre attiva ed ha 4 livelli di luminosità regolabile
- dal punto di vista estetico il case è più brillante, le casse occupano meno spazio e la posizione di tasti e microfono è stata modificata
- la pennina touch è diventata più grande
- a causa delle dimensioni più ridotte le cartucce del GBA sporgono dal sistema Lite; per questo, al fine di dare una sensazione di continuità dei profili, è inclusa una cartuccia che fa da "filler" quando non ci sono giochi GBA inseriti nell'apposito slot.

Il suo product code è NTR-001 (USG-001 per la versione Lite) e rappresenta la contrazione del suo code name di sviluppo "Nitro"; il significato del suffisso DS sta per

"Developers System" ma anche per "Dual Screen".

Come Nintendo ci ha abituato, anche questa console è retrocompatibile con i giochi del suo predecessore portatile, il GBA, ed il boot può essere eseguito o da una cartuccia NDS, o da una cartuccia GBA, o dal Firmware oppure via WiFi. Quando è inserita una cartuccia GBA (32pin) l'NDS automaticamente swtcha in GBA Mode e diventa "quasi" come un GBA vero.

Le differenze con un vero GBA sono le seguenti:

- Non supporta le cartucce GB/GBC
- Il registro non documentato Internal Memory Control (Port 800h) non è supportato quindi l'NDS non permette di utilizzare la RAM overlocked
- non esiste una link port (quindi manca il supporto per l'hardware link port) di conseguenza i giochi GBA possono essere utilizzati solo in modalità 1 player
- la frequenza audio sembra passare da 16.78MHz a 16.76MHz, quindi lievemente più lenta dell'originale GBA (differenza impercettibile ma a quanto pare documentata)
- Nel BIOS GBA cambia un byte che passa dal valore 00 a 01 facendo cambiare il checksum dello stesso
- Il logo GBA puo'essere fatto comparire o nello schermo superiore o nell'inferiore (opzione selezionabile nel menu di avvio)
- Lo schermo di gioco GBA ha delle bande nere laterali come nella foto sottostante:



Il motore del DS è un processore ARM9 che collabora con un ARM7 probabilmente per motivi legati ad una più agevole retrocompatibilità con il GBA che è appunto dotato di un ARM7.

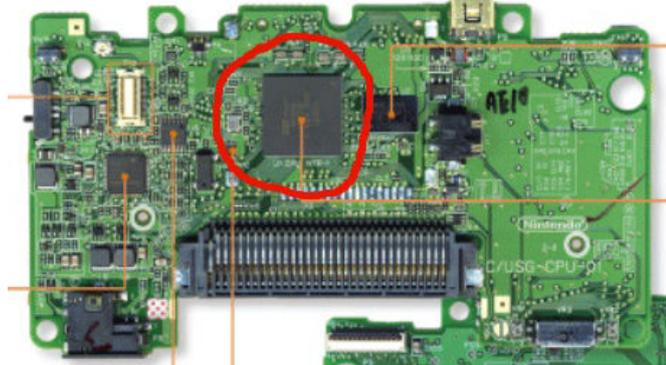




LE PROTEZIONI

BIOS

Il BIOS del ARM7 (chiamato NDS7), di circa 45KB, è suddiviso in 2 regioni: la prima che viene esclusa dalla lettura, tramite la scrittura del registro 1024h, subito prima della lettura della cartuccia di gioco come per il GBA ed un'altra invece sempre attiva.



La parte protetta contiene il cartridge KEY1 encryption seed utilizzato per codificare/decodificare i dati memorizzati su cartuccia. Inoltre le funzioni proprie del BIOS hanno una protezione software che impedisce loro di leggere l'area protetta.

Il sistema per leggere i dati protetti è simile a quello utilizzato per il GBA e cioè utilizzare dei tricks hardware per leggere la memoria prima che venga protetta; in alternativa sono stati scoperti 2 opcodes (THUMB opcodes 05ECh e 05EEh) capaci di bypassare questo vincolo.

Il BIOS del ARM9 (NDS9), di circa 65KB, non è protetto dalla lettura nè via hardware nè via software.

Il BIOS verifica l'header della cartuccia (header che va dall'offset 0x0000 al 0x0200), in particolare controlla il checksum del logo Nintendo memorizzato all'offset 0x15C (checksum calcolato con algoritmo CRC-16 dall'offset 0x00C0 all'offset 0x015B) e corrisponde ad un valore fisso = CF56

Il firmware successivamente controllerà anche i bytes del logo stesso (il logo Nintendo va dall'offset 0x00C0 all'offset 0x015B).

IL FIRMWARE

Il firmware è memorizzato all'interno di un chip NVRAM che ha dimensioni differenti a seconda del tipo di console e del bundle:

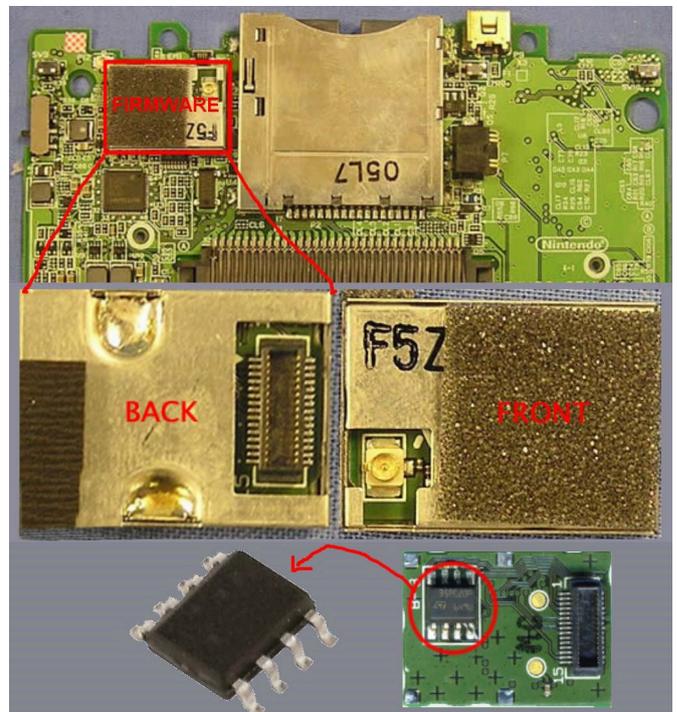
- ST M45PE20 - 256 KBytes (Nintendo DS)
- ST M35PE20 - 256 KBytes (Nintendo DS-Lite)
- ST M25PE40 - 512 KBytes (iQue DS, conc aratteri cinesi)
- ST 45PE40V6 - 512 KBytes (DS Zelda, NTR-AZEP-0)

- ST 45PE80V6 - 1024 Kbytes (es. Spirit Tracks, NTR-BKIP)
- Sanyo LE25FW203T - 256 KBytes (MarioKart backup)

In ogni caso il "cuore" è di 256KB ed è suddiviso in 5 sezioni criptate e compresse:

- part1 - ARM9 BOOT CODE
- part2 - ARM7 BOOT CODE
- part3 - ARM9 GUI CODE
- part4 - ARM7 GUI CODE
- part5 - DATA/GFX

che vengono verificate con dei checksums. Il coder Chishm è stato in grado di reversare la compressione/cryptazione ed ha sviluppato un tool per estrarre le varie parti del firmware; il tool si chiama Fwunpacker. Esiste inoltre uno strumento chiamato CFW Suite per elaborare un firmware e modificarne la GUI.



La possibilità di agire a così basso livello nel sistema ha permesso inoltre la creazione di un virus in grado di brickare la console, a quanto pare funzionante solo se si era effettuata la softmod, mostrando un muro di mattoni (brick wall) nel dual screen all'avvio:



E' però possibile effettuare un "hot swap" del chip contenente il firmware permettendo così di recuperare delle console brikkate flashando un firmware valido con FlashMe.





CARTUCCE IN FORMATO PROPRIETARIO



Le cartucce utilizzate dal DS hanno un formato proprietario con 17 PINs ed hanno product code NTR-005; ne esiste una variante con supporto di hardware infrarossi con codice NTR-031:



Come per il GBA anche per queste cartucce esistono i cloni (reale a destra, fake a sinistra):



Per ulteriori dettagli su come riconoscerle andate qui.

DATI CRIPTATI

I dati su cartuccia possono essere criptati. In particolare nel range 0x4000-0x7FFF di una cartuccia esiste una "SECURE AREA" accessibile al solo ARM9 e da lui decriptabile attraverso comandi criptati utilizzando il KEY1 seed; non tutte le cartucce hanno la secure area; una volta decriptata, solamente i primi 8 bytes (che contengono il Secure Area ID) di questo range vengono controllati; se il check è corretto il caricamento della cartuccia prosegue.

I giochi caricati via WLAN non hanno la secure area ed i giochi che la hanno non possono essere avviati via WLAN. L'algoritmo di criptazione è basato sul "Blowfish Encryption Algorithm" ideato da Bruce Schneier nel 1993.

Le cartucce sono comunque dumpabili attraverso specifici comandi del protocollo SPI; i chips dumpabili di una cartuccia sono i seguenti:

Type / Total Size / Page Size / Chip-Example / Game-Example
EEPROM 0.5K bytes 16 bytes ST M95040-W (eg. Metroid Demo)

EEPROM 8K bytes 32 bytes ST M95640-W (eg. Super Mario DS)

EEPROM 64K bytes 128 bytes ST M95512-W (eg. Downhill Jam)

FLASH 256K bytes 256 bytes ST M45PE20 (eg. Skateland)

FLASH 256K bytes Sanyo LE25FW203T (eg. Mariokart)

FLASH 512K bytes 256 bytes ST M25PE40? (eg. which/any games?)

FLASH 512K bytes ST 45PE40V6 (eg. DS Zelda, NTR-AZEP-0)

FLASH 1024K bytes ST 45PE80V6 (eg. Spirit Tracks, NTR-BKIP)

FLASH 8192K bytes MX25L6445EZNI-10G (Art Academy only, TWL-VAHV)

FRAM 8K bytes No limit ? (eg. which/any games?)

FRAM 32K bytes No limit Ramtr

Nota: le FRAM sono memorie RAM che mantengono dati permanentemente senza necessitare di una batteria (al contrario delle SRAM) e sono più veloci delle EEPROM, con le quali sono comunque retrocompatibili.

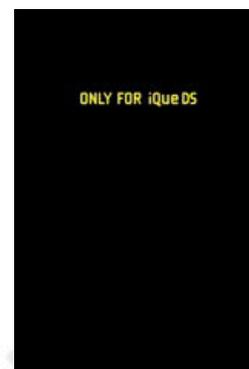
Una volta avvenuto il riconoscimento tramite i comandi criptati con la KEY1 i dati vengono scambiati attraverso un altro algoritmo chiamato KEY2 basato su 2 registri da 39 bits, degli XOR e degli shift.

Da notare che TUTTI i dumps delle cartucce NDS che si trovano online hanno l'header decriptato.

REGION CHECK

La console non ha alcun region check.

Le cartucce di gioco iQue DS contengono invece una funzione che controlla la regione della console e se non la trovano non fanno partire il gioco (tale protezione è dunque nel gioco, non nella console).



Per ovviare al problema, nelle consoles DS, va editato l'offset 0x1D della ROM e cambiato il valore 80 con il valore 00:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
00000000	4E	49	4E	54	45	4E	44	4F	47	53	00	00	00	41	33	36	43	NINTENDOGS...A36C
00000010	30	31	00	00	08	00	00	00	00	00	00	00	00	00	00	00	00	01.....
00000020	00	40	00	00	00	08	00	02	00	00	00	02	D8	C0	0B	00	000A..
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00



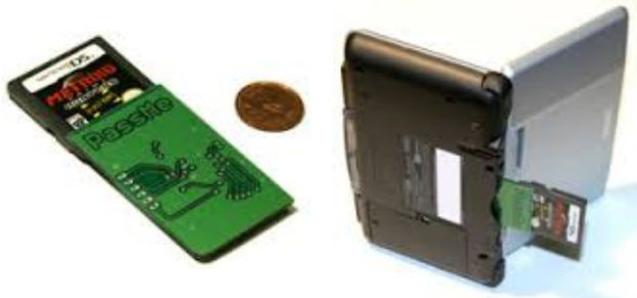


SISTEMI PER ESECUZIONE DI CODICE ARBITRARIO

FIRMWARE DA V1 a V3

Prima che nel 2006 venisse crackato l'algoritmo di criptazione, per avviare codice non criptato esistevano vari sistemi che sfruttavano diversi mancati controlli da parte del firmware della console:

PassMe



E' un hardware che si interpone tra la console e la cartuccia nello slot GBA (SLOT-2), legge l'ID da quest'ultima aggirando il sistema facendogli pensare che la cartuccia flash inserita sia originale. Era la base per tutte le modifiche software alla console.

WiFiMe

Il DS ha la possibilità di collegarsi via WiFi e di inviare anche giochi tramite WiFi!

Il servizio DS Download Play del firmware V1 permette infatti di inviare applicazioni ed una falla nella gestione della firma RSA originale di tale sistema ha permesso agli hackers di creare WiFiMe, un programma per avviare gli homebrew.

Nel dettaglio la prima versione del titolo Download Play utilizza un indirizzo di esecuzione da un header che non fa parte del calcolo della firma digitale del codice e dei dati. WiFiMe è stato estratto da Super Mario 64 DS con l'indirizzo di esecuzione modificato nello spazio GBA ROM, ma sarebbe potuto essere stato qualsiasi programma DS Download Play.

Wireless Multiboot (WMB)

Questa applicazione per Windows creata dal coder FireFly necessita d una scheda di rete PCI o PCMCIA con chipset Ralink; è in grado di inviare piccoli programmi firmati Nintendo al DS Download Play.

FlashMe

Questo sistema sovrascrive il firmware del NDS permettendo l'esecuzione immediata di codice DS anche nel GBA ROM space, rimuovendo il check della firma digitale per il DS

Download Play e permettendo così l'invio di homebrew. Siamo di fronte ad una delle primissime softmod per consoles.

FIRMWARE V4

Dalla versione firmware V4 i bugs che permettevano i sopracitati exploit sono stati patchati ma questo non ha scoraggiato gli hackers che sono andati avanti nella classica rincorsa del gatto col topo.

Come controllare la versione del firmware del NDS:

Ci si basa sul comportamento dell'applicazione PictoChart: si avvia il DS con un gioco inserito, si avvia PictoChart e si estrae la cartuccia a console ancora accesa; apparirà un colore degli schermi che è diverso per ogni versione di firmware (questa tecnica viene anche chiamata "Pink Screen of Death" o "TakeMeOut"):

v1: PictoChat si blocca/non viene mostrato alcun colore

v2: due schermi grigio-blu

v3/iQue: due schermi verde scuro

v4: due schermi giallo oro

v5: due schermi magenta (i DS lite escono con questa versione, assieme ad alcuni DS rosa):



v6: due schermi blu scuri

v7: il sistema non va in crash (solo nella versione dorata del DS Lite limited edition giapponese).

PassMe2

Come il PassMe ma sfruttando un diverso bug del BIOS DS utilizzando giochi specifici (una lista di quelli supportati è disponibile qui)

FlashMe v5

In questa versione il codice del programma viene ridimensionato e scritto in aree protette del firmware che





non possono essere riscritte da alcun gioco originale. Le precedenti versioni di FlashMe erano infatti scritte in aree che si supponevano essere "inutilizzate" scoprendo in seguito che non era così.

NoPass

Dal momento in cui nel 2006 l'algoritmo di criptazione delle cartucce fu reversato da Martin Korth (autore del emulatore No\$GBA nonché colui che per primo riuscì anche a dumpare il bios del NDS) si riuscì ad eseguire codice anche da cartucce inserite nello SLOT-1 (quello per i giochi NDS); questo metodo prese il nome di "NoPass".

Cartucce NoPass

Iniziarono dunque ad uscire cartucce che non necessitavano di un gioco originale per eseguire codice arbitrario perchè erano codificate proprio come le cartucce originali come ad esempio la Max Media Launcher di Datel che nella foto qui sotto potete vedere a confronto con un più datato PassMe:



A seguire poi arrivarono le decine e decine di R4, M3, AceKard, ecc.



ognuna con i propri firmwares di avvio alcuni dei quali anche reversati come quello della R4 sempre grazie a Chishm ed al suo tool r4crypt (esiste anche injektor2 1.08 - ultima versione 1.09 all'apparenza introvabile).

SOUNDHAX

A fine 2014 il reverser/dev/coder Smealum ha mostrato un video dell'hack basato su segnale audio per il gioco Bangai-o-Spirit; il gioco permette di trasferire livelli via audio: reversando il codice è stato possibile effettuare un buffer overrun ed eseguire codice arbitrario semplicemente mandando in play un .WAV (è possibile vederlo all'opera anche nell'emulatore DeSmuME).

CONCLUSIONI

Come potete notare le cose iniziano a diventare molto interessanti nel mondo del console hacking soprattutto perchè per la prima volta iniziamo ad avere la possibilità di scrivere arbitrariamente nella memoria interna di una console così da avere una sorta di primissimo e rudimentale custom firmware. Abbiamo inoltre visto come siano state messe per la prima volta in atto contromisure volte a limitare i danni provocati dai bugs scoperti dai reverser: gli oggiogiorno temutissimi FIRMWARE UPDATES ! Da notare che il firmware della console non poteva aggiornarsi nè tramite WiFi nè tramite cartuccia quindi quando si acquistava la console si andava un po' a "fortuna" quando ancora non era noto il metodo del "Pink Screen" spiegato più in alto. Anche se molto tardivi ci sono anche exploit che sfruttano falle nei giochi come l'exploit SoundHax. Mi scuso se posso essere sembrato sbrigativo su alcuni argomenti ma il DS era una console a me piuttosto sconosciuta ed ho dovuto studiarla praticamente da zero. Se avete domande o precisazioni scrivetemi.

La prossima puntata parlerà dell'evoluzione di questa Nintendo sia in termini di hardware che in termini di sicurezza.

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.





Atari 5200 SuperSystem

di Leonardo Miliani

Nel 1982 Atari, allora indiscussa dominatrice del mercato videoludico domestico, presenta l'erede della 2600/VCS, quella console che da alcuni anni sta letteralmente facendo il bello ed il cattivo tempo tra i sistemi elettronici da gioco, presente in milioni di case. Messa in commercio come diretta concorrente del Mattel Intellivision, la nuova console non solo non riuscirà a tener testa a quest'ultimo ma cederà il passo anche al ColecoVision, che arriverà poco prima del suo debutto, e alla fine verrà tolta dal commercio addirittura prima ancora della console di cui avrebbe dovuto prendere il posto. Stiamo parlando della sfortunata Atari 5200 SuperSystem, che avrebbe meritato più fortuna.

Le origini

Nel 1975 Atari decide di sviluppare un dispositivo elettronico in grado di riprodurre su un comune televisore i suoi giochi arcade. A differenza del Pong, che è un dispositivo basato su componenti discreti e integrati TTL, e quindi può eseguire solo il gioco per cui è stato progettato, Atari vuole che il nuovo dispositivo sia "programmabile", ossia che si possa cambiare in modo facile il gioco da riprodurre. A metà di quell'anno MOS Technology presenta la CPU 6502, un microprocessore economico che apre la strada alla realizzazione della console: Atari tratta infatti per la fornitura di una versione depotenziata del chip e immette sul mercato la VCS, da "Video Computer System". La console deve molte delle sue fortune anche ad un particolare chip che affianca il processore ed a cui viene affidato non solo il compito di generare l'audio ma anche quello di renderizzare l'immagine sullo schermo e muovere degli speciali elementi grafici, gli "sprite", che rappresentano il giocatore, i personaggi e gli oggetti presenti sulla scena: questo chip si chiama TIA, sigla che sta per "Television

Interface Adaptor".

A capo dello sviluppo del TIA c'è Jay Miner (fig. 2), che sarà in seguito conosciuto come il "padre dell'Amiga", per il contributo che darà allo sviluppo del chipset dell'omonima piattaforma a 16 bit che prenderà vita qualche anno dopo. L'idea di Miner è tanto semplice quanto efficace: delegare compiti gravosi a chip specifici, o co-processori, così da alleggerire il lavoro della CPU. Il TIA viene progettato proprio per togliere al microprocessore della VCS la gestione dell'audio e del video. La console viene completata nel 1976, anche se viene messa in vendita solo nel 1977. Quasi subito, parte lo sviluppo della sua erede: tanta fretta è dettata dal fatto che la vita commerciale della VCS viene stimata dai suoi stessi creatori in circa 3 anni (quindi arriviamo al 1979/1980), anche alla luce delle scelte progettuali fatte per contenerne il costo di produzione che si sono riflesse su un hardware non proprio di ultimo grido e che è previsto vada incontro ad una rapida obsolescenza. Nonostante ciò, la VCS vende bene e resta sulla cresta dell'onda ben oltre questo lasso di tempo, non certo però per le sue caratteristiche tecniche bensì per il ricco parco software che offre ai suoi utenti (potete approfondire l'argomento sull'articolo ad essa dedicato sullo scorso numero della nostra rivista).

Ignari del fatto che la VCS durerà più a lungo di quanto previsto, gli sviluppatori Atari iniziano a lavorare ad un progetto a cui viene assegnato il codice "3200". Le caratteristiche devono essere ben superiori a quelle della VCS per cui viene deciso di utilizzare una CPU a 10 bit a cui affiancare diversi chip personalizzati, ognuno dedicato ad uno specifico compito. Il gruppo capitanato da Jay Miner si concentra su un chip grafico erede del TIA, denominato CTIA, da "Color Television Interface Adaptor", al cui sviluppo partecipano Doug Neubauer, Steven Mayer e Joe Decuir. Il CTIA, derivando dal TIA, presenta gli stessi limiti di quest'ultimo: non è stato progettato per gestire framebuffer, grafica bitmap e molteplici modi grafici. Joe Decuir, Francois Michel e Steve Smith, sempre sotto la direzione di Miner, lavorano ad un altro chip da affiancare al CTIA, l'ANTIC, da "Alphanumeric Television Interface Controller", un vero e proprio processore grafico programmabile deputato alla generazione dell'immagine vera e propria, capace di accedere in autonomia alla memoria ("Direct Memory Access", o DMA) per prelevare i dati che gli occorrono, e che può gestire diversi modi



Fig. 1: Atari 5200 SuperSystem
(foto: Evan-Amos - fonte: Wikimedia Commons)





Fig. 2 - Jay Miner (fonte: Alcheron.com)

grafici, compresi quelli testuali. Affianca questi chip un integrato dedicato alla generazione del suono ed al controllo dei sistemi di input denominato POKEY, da "Pot Keyboard Integrated Circuit", al cui sviluppo contribuiscono Steven Mayer e Ronald Milner. Il nuovo hardware è abbastanza complesso e i lavori procedono a rilento. Come se ciò non bastasse, verso la fine del 1977 parte la rivoluzione degli home computer: vengono messi in commercio 3 piccoli computer che sconvolgono il mercato informatico, vale a dire il Commodore PET 2001, l'Apple II e il TRS-80. Attratti dalle possibilità che un computer offre in più rispetto ad una semplice console, vale a dire non solo la possibilità di giocare ma anche quella di usare programmi di vario genere, e considerato il prezzo d'acquisto non più proibitivo che li rende strumenti

finalmente alla portata di tutti, molti acquirenti si orientano verso questi nuovi prodotti. Atari decide di fermare lo sviluppo della nuova console e di entrare anch'essa nel mercato degli home computer: per non perdere troppo terreno rispetto alla concorrenza e per non gettare alle ortiche tutto il lavoro di sviluppo fatto fino ad allora, decide di riciclare quanto già pronto. C'è un problema sulla parte video: il CTIA non è ancora del tutto sviluppato e mancano alcune modalità grafiche previste dal progetto ma, per non rimanere troppo indietro, Atari lo mette lo stesso in produzione così com'è, sostituendolo poco dopo con la versione completa, denominata GTIA, da "Graphic Television Interface Adaptor" (fig. 3). Il tutto viene assemblato intorno ad una CPU derivata dal MOS 6502 e poi messo in vendita nel 1979 come due computer simili ma con alcune differenze: stiamo parlando dell'Atari 400 ed 800.

Il nuovo corso

Dopo il debutto di queste macchine, il progetto della console resta fermo ancora per un certo lasso di tempo: le vendite dei due computer vanno molto bene, ed ancora meglio sta andando la VCS, che sta stracciando ogni record, con le vendite che raddoppiano anno dopo anno. Nonostante questi numeri, Atari teme la concorrenza: Mattel, agli inizi del 1980, mette in commercio l'Intellivision, una console tecnicamente superiore alla VCS, la quale inizia a mostrare tutta la sua arretratezza tecnologica. Mattel sa che quello è il lato debole della VCS e spinge proprio sulla comparazione tra la resa dei giochi sulla sua macchina e quella del modello Atari: sull'Intellivision la grafica e l'audio si avvicinano a quelli dei giochi da bar, mentre sulla VCS la grafica è resa in modo rudimentale ed il sonoro è limitato a semplici toni, rendendo molto difficile la conversione degli arcade più recenti. La conversione di Pac-Man è rappresentativa di questi limiti: le differenze hardware ci sono e si vedono tutte nel gioco di Atari, solo lontanamente paragonabile all'arcade



Fig. 3 - L'Atari GTIA (foto: Rodrigo Castro - fonte: Wikimedia Commons)





originale. Viene perciò ripreso in mano il progetto di "Pam", come era stata chiamata affettuosamente la console dai suoi sviluppatori, nome che sembra derivi da quello di un'impiegata Atari. Dai vertici aziendali viene però presa una decisione che, alla lunga, si rivelerà un vero e proprio autogol: nonostante l'hardware sia identico a quello dei computer 400/800, si decide di rendere incompatibili le due piattaforme così che le cartucce per gli home computer non possano essere inserite e giocate sulla console e viceversa. La console viene dotata di 4 porte giochi per altrettanti joystick e, sulla parte superiore, della porta per l'introduzione delle cartucce. Il corpo macchina è realizzato in plastica nera con una fascia metallica che attraversa orizzontalmente la parte superiore e, posteriormente, uno sportello che nasconde alla vista un alloggiamento dove riporre i 2 joystick in dotazione: proprio la presenza di questo vano è il motivo per cui la macchina appare di generose dimensioni. Internamente la scheda madre è basata sulla stessa CPU e sul tritico di processori ausiliari presenti sugli home computer Atari: la CPU è un 6502 modificato per lavorare a 1,79 MHz, internamente noto come SALLY, mentre la grafica è gestita dai 2 chip GTIA e ANTIC, ed il suono e l'input dal POKEY. Vengono sviluppati anche dei particolari joystick dotati di un tastierino numerico per l'introduzione di comandi addizionali, per l'avvio ed il reset dei giochi e, prima fra

tutte le altre console, anche di un pulsante per la messa in pausa degli stessi.

Durante lo sviluppo, viene inizialmente scelto di commercializzare la console con il suo nome di progetto Pam ma trasformandolo in PAM, acronimo inverso per "Personal Arcade Machine". Successivamente viene però cambiato in "Atari Video System X" e diversi prototipi finali della macchina vengono effettivamente realizzati con questo nome (qualcuno si trova ancora in circolazione). Alla fine viene però deciso di utilizzare la nomenclatura numerica adottata per gli home computer, e di ribattezzare nel contempo anche la VCS: quest'ultima diventa l'Atari 2600, perché durante il suo sviluppo quello è stato il suo codice interno, ed il Video Sistem X diventa l'Atari 5200. Questo numero non è scelto a caso: essendo il doppio di 2600, esso vuole indicare che la console è due volte più potente rispetto alla precedente. Il debutto avviene nel mese di novembre del 1982 quando la console viene presentata come Atari 5200 SuperSystem e messa in vendita al prezzo di 269,99 dollari.

Caratteristiche tecniche

La console è composta da un grande corpo in plastica contenente la scheda madre e, come detto, un alloggiamento dove riporre i 2 joystick in dotazione. Internamente abbiamo la stessa CPU degli home computer, ossia il MOS

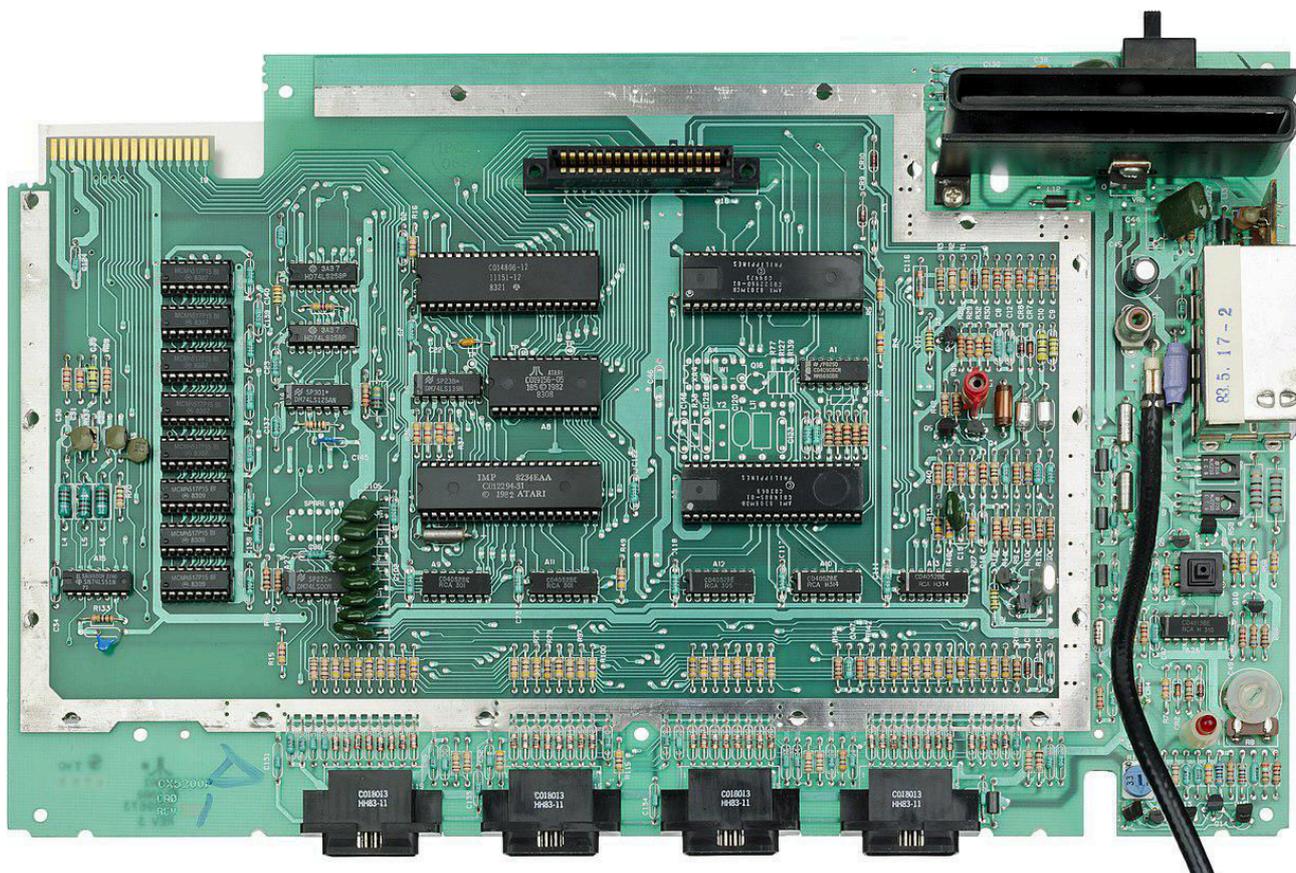


Fig. 4 - Uno dei primi prototipi del 1975 della futura VCS/2600 (fonte: Wikimedia - autore: Chris Platsikoudis)





6502C, dove la "C" non indica una revisione successiva ma solo che questo è un chip "custom" capace di operare a 1,79 MHz. Il reparto video è gestito dall'accoppiata GTIA/ANTIC. Il GTIA è la versione completa del progetto originale del CTIA, a cui aggiunge 3 modalità grafiche aggiuntive. L'ANTIC è il chip deputato alla generazione della grafica bitmap. Esso è un vero e proprio processore programmabile mediante mini-programmi chiamati "display list": come un comune interprete BASIC, il processore legge la display list ricevuta e genera ogni riga dello schermo in base alle istruzioni ivi contenute. L'ANTIC recupera i dati direttamente dalla memoria mediante un accesso diretto senza l'intermediazione della CPU (modalità DMA) e poi passa i dati dell'immagine generata al GTIA. Quest'ultimo applica i colori gestendone anche le tonalità: grazie a questo modo di operare, si possono avere 16 tonalità per i 16 colori base, per un totale di 256 colori. Il GTIA applica anche gli sprite all'immagine: come nel TIA, questi sono denominati "Player/Missile" e sono, rispettivamente, 4 "player" larghi 4 pixel l'uno e 4 "missile" larghi 2 pixel l'uno. Entrambi hanno un'altezza pari a quella dell'immagine video. Le modalità video sono nel complesso 14: ci sono 6 modi testuali ed 8 modi grafici, questi ultimi con una larghezza di 80, 160 o 320 pixel. L'altezza può arrivare fino a 192 pixel anche se, in overscan, l'immagine può raggiungere dimensioni massime di 384x240 pixel. I colori visualizzabili contemporaneamente sullo schermo sono 2, 4 oppure 16 rispettivamente per immagini larghe 320, 160 od 80 pixel. Siccome l'ANTIC, tramite le display list, può cambiare i valori dei registri ad ogni riga dell'immagine, si possono avere fino a 256 colori diversi (16 colori per 16 tonalità) e 16 colori diversi per ogni riga. È presente anche lo scrolling sia orizzontale che verticale.

Il POKEY si occupa primariamente della generazione sonora. Questo chip offre ben 4 canali differenti, che possono operare sia come 4 canali ad 8 bit che come 2 canali a 16 bit. Ogni canale permette di gestire in maniera indipendente sia il volume che le caratteristiche dell'onda generata (onda quadra con "duty cycle" variabile oppure rumore). Oltre a ciò, il POKEY si occupa anche della lettura dei dispositivi di input.

La console è dotata di 16 KB di memoria RAM, di cui le prime due pagine usate dalla CPU e dal BIOS della console, ed il resto a disposizione del gioco in cartuccia. Relativamente al BIOS, questo risiede su 2 KB di ROM: il primo KB contiene la mappa caratteri (identica a quella degli Atari 400/800) mentre l'altro è il BIOS vero e proprio, qui indicato come "monitor". Questo è composto dalle routine di inizializzazione della macchina eseguite prima di avviare il gioco su cartuccia, da quelle per la gestione degli interrupt e dal



Fig. 5 - Il controller con il comando capace di movimento a 360° (foto: Evan-Amos - fonte: Wikimedia Commons)

sistema di "shadowing" dei registri interni della macchina. La console può indirizzare al massimo cartucce con 32 KB di ROM: quantitativi superiori richiedono l'utilizzo della tecnica del "bank switching", ossia un meccanismo che seleziona, tra i vari banchi, quello attualmente in uso.

I joystick sono il punto distintivo di questa macchina nonché il suo tallone di Achille (fig. 5): sviluppati per avere un movimento a 360° e non essere limitati alle sole 4/8 direzioni dei dispositivi degli altri sistemi, non sono però dotati del riposizionamento automatico. Ciò significa che, una volta cessata la pressione per azionarli, l'asta non recupera la posizione centrale: questa cosa crea notevoli problemi in alcuni giochi, tipo ad esempio Pac-Man, dove il comando continua a mandare il personaggio in una data direzione se non viene rimesso nella zona neutra. Oltre a ciò, vengono scelti dei materiali scadenti per cui sono frequenti le rotture delle plastiche, con la necessità di dover ricomprare l'intero joystick.

Come detto in precedenza, Atari ha apportato delle modifiche all'hardware affinché i giochi dei computer 400/800 non siano utilizzabili sulla 5200 e viceversa. Quest'impossibilità è primariamente da imputare al BIOS ridotto da 10 ad 1 KB, per cui le funzionalità integrate nei computer non sono presenti nella console. Anche i registri del GTIA e del POKEY sono mappati a indirizzi diversi, perciò la conversione di un gioco è possibile ma vanno cambiati manualmente tutti i valori affinché il software ritrovi questi registri alle giuste locazioni. Come se non bastasse, alcuni di questi registri hanno nella 5200 un compito differente rispetto a quelli del computer. Infine, i particolari joystick della console sono visti come una coppia di paddle che forniscono segnali analogici richiedendo





una gestione del loro input completamente diversa rispetto a quella dei joystick digitali usati sui computer Atari.

Una particolarità della console è la presenza dello Switchbox, un commutatore automatico per il segnale della console da collegare dietro al televisore di casa (fig 6). Normalmente le altre console coeve offrono un commutatore manuale: è l'utente che deve spostare l'interruttore per selezionare l'ingresso della console oppure del cavo TV. In quello della 5200 un particolare circuito elettronico, non appena si accorge della presenza del segnale video sull'ingresso della console, commuta su quest'ultima. Per ridurre il numero di cavi, quello di collegamento fra televisore e console porta anche l'alimentazione a quest'ultima.

Giochi, accessori e versioni

La console, avendo avuto una vita commerciale abbastanza breve, non ha ricevuto molto interesse da parte degli sviluppatori. Nei 2 anni in cui è rimasta in commercio sono stati presentati solo 69 giochi, per lo più sviluppati da Atari stessa, e principalmente conversioni di famosi giochi da bar come "Defender", "Frogger", "Galaxian", "Ms. Pac-Man" (fig. 8), "Popeye", "River Raid", "Space Invaders", "Zaxxon" (fig. 9). Insieme alla console viene commercializzato anche uno speciale controller denominato "Pro-Line Track Ball" (fig. 7), dotato di doppia tastiera alfanumerica e track-ball centrale, pensato per giochi come "Centipede" e "Missile Command". A differenza di Coleco, Atari non ha mai messo in commercio un accessorio per trasformare la console in computer, probabilmente per via del fatto che ciò avrebbe intaccato le vendite dei suoi computer della serie 400 ed 800. La limitata diffusione

si è ripercossa anche sullo scarso interesse dei produttori di hardware, tanto che poche sono le alternative ai suoi particolari joystick.

Nel 1983, dopo un solo anno di commercializzazione, Atari presenta una revisione della console. Esteriormente il nuovo modello si differenzia per la presenza di 2 sole porte joystick e per la scomparsa del commutatore automatico, rimpiazzato da uno manuale. Questa modifica comporta anche lo spostamento dell'alimentatore all'interno della console stessa. Insieme alla nuova console Atari presenta anche un adattatore per poter riprodurre le cartucce della 2600, cosa che Coleco ha già fatto in precedenza per il suo ColecoVision (attirandosi le ire di Atari). Nonostante ciò, le vendite non vanno molto bene per cui, agli inizi del 1984, Atari ne annuncia la dismissione durante la conferenza stampa di presentazione della sua erede, la 7800 (che però, per via della divisione e riorganizzazione della casa madre, non arriverà sugli scaffali che nel 1986).

Il fallimento

Le ragioni dell'insuccesso della console sono molteplici, e non ce n'è una predominante. Probabilmente il principale fattore può essere stato il ritardo nella sua commercializzazione: nonostante le discrete capacità hardware, la console era nata per debuttare almeno un paio di anni prima, e la pausa nel suo sviluppo l'ha fatta uscire sul mercato già "vecchia". L'incompatibilità con i giochi dei computer Atari 400/800 non ha certamente aiutato visto che piattaforme assolutamente identiche sono state isolate l'una dall'altra con illogiche scelte



Fig. 6 - L'Atari Switchbox, il commutatore TV automatico (foto: Evan-Amos - fonte: Wikimedia Commons)





Fig. 7 - Il Pro-Line Track-Ball (foto: Evan-Amos - fonte: Wikimedia Commons)

progettuali rivelatesi, alla fine, vere e proprie zappate sui piedi.

Un altro duro colpo è stato dato dall'uscita del ColecoVision, avvenuta a metà del 1982, proprio qualche mese prima del debutto della 5200: quest'ultima doveva competere con l'Intellivision di Mattel ma si è ritrovata con un avversario molto più avanzato tecnologicamente che le ha tolto ulteriori potenziali acquirenti. Un'altra critica che le è stata mossa è la discutibile scelta di includere nella confezione di vendita l'adattamento di "Super Breakout", un gioco ormai datato e certamente non in grado di mostrare le capacità della console: il confronto con "Donkey Kong" offerto insieme al ColecoVision è impietoso ed a tutto vantaggio di quest'ultima macchina, dove il gioco è riprodotto in maniera quasi del tutto identica rispetto alla versione arcade. Inoltre Coleco capisce anche che la 2600 non è facile da far fuori e preferisce offrire fin da subito un adattatore per permettere di usare le sue cartucce con il ColecoVision, offrendo in questo modo ai possessori della console Atari la possibilità di non doversi disfare di tutti i giochi di cui sono già in possesso. Atari, dal canto suo, non ha capito subito questo

punto ed ha offerto un tale adattatore solo 1 anno dopo la commercializzazione della 5200, quando la 2600 registrava i suoi massimi volumi di vendita.

Un altro punto a sfavore della 5200 è stato il suo elevato costo: il suo prezzo iniziale di vendita di 269,99 dollari non è certamente allettante se paragonato ai 179 dollari del ColecoVision. Ma non regge neanche nei confronti della 2600, venduta a 79 dollari e dotata non solo di migliori controller ma anche di un parco giochi composto da centinaia di titoli. Infine, è arrivata la crisi dei videogiochi del 1983 a dare il colpo finale ad una console che non ha mai brillato. Si stima che nei 2 anni di commercializzazione, la 5200 sia stata venduta in circa 1 milione di esemplari, lo stesso numero di unità che la 2600 ha fatto registrare nel solo 1979. Infine, una nota sulle aree di commercializzazione. La 5200 è stata realizzata nella sola versione NTSC, nonostante i chip video ANTIC e GTIA siano stati prodotti anche in versione PAL per i computer Atari destinati al mercato internazionale: a causa di ciò la console ha avuta una ulteriore limitazione alla sua diffusione, circoscritta al solo mercato nordamericano (USA e Canada), che si è ripercossa negativamente sulle vendite.

Peccato, perché le potenzialità c'erano tutte, ma un prezzo troppo elevato, alcune discutibili scelte progettuali, un'uscita ritardata, la stessa Atari che non ci ha creduto troppo convintamente ed ha continuato a supportare una console ormai obsoleta ma fonte di sicuri guadagni, ed un misero supporto da parte delle software house l'hanno ridotta a semplice comprimaria e come tale il suo rapido passaggio non ha lasciato una traccia significativa nel panorama videoludico.

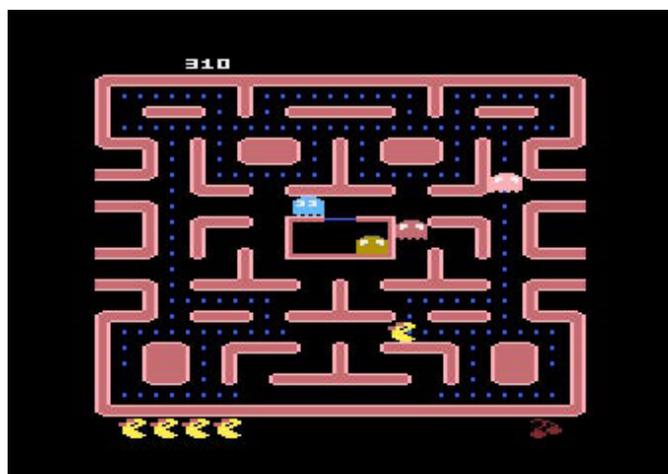


Fig. 8 - Ms. Pac-Man (fonte: Mobygames.com)



Fig. 9 - Zaxxon (fonte: Mobygames.com)





CALC RESULT - lo Spreadsheet per Commodore 64

di Francesco Fiorentini

Calc Result é uno spreadsheet versatile e potente prodotto dalla **Handic Software AB** nel 1983.

All'epoca erano state create addirittura due versioni per venire incontro alle esigenze economiche degli utenti:

C.D	CALC RESULT EXPANDED (in italiano) Potente programma di pianificazione manageriale.	C 64, SX 64	195.000
C	CALC RESULT EASY (in italiano)	C 64, SX 64	125.000

Le caratteristiche di questo foglio di calcolo sono: 32 pagine, suddivisione orizzontale e verticale delle pagine, formule, testo, grafici, finestre, calcolo automatico, messaggi di errore, schermate di aiuto multilingue, memorizzazione in file DIF, comandi di sistema, disco, modifica, stampa e formato.

P	A	B	C	D
1	Agente	Marca	Q.ta	Prezzo
2	Roger	BMW	12	33000
3	Franco	AUDI	13	35000
4	Lupo	FIAT	20	17000
5	Marco	Ferrari	5	150000
6	Jimmy	Porsche	4	125000
7	George	Jaguar	7	68000

P	Marca	Q.ta	Prezzo	Totale
2	BMW	12	33000	396000
3	AUDI	13	35000	
4	FIAT	20	17000	
5	Ferrari	5	150000	
6	Porsche	4	125000	
7	Jaguar	7	68000	

Handic Software AB

Handic Software AB, fondata nel 1952, era originariamente un'azienda produttrice di radio e telefoni. Datatronic acquisì l'azienda per una corona svedese nel 1980 e utilizzò il marchio come distributore ufficiale dei computer Commodore in Scandinavia. Oltre ai computer, l'azienda iniziò a importare titoli software internazionali e nel 1983

pubblicò il primo gioco commerciale sviluppato in Svezia, Space Action. Handic Software AB chiuse i battenti nel 1986 dopo che Commodore aprì un suo ufficio in Svezia.

Calc Result, uno dei prodotti di Handic Software, é un foglio di calcolo veramente ben fatto ed estremamente attuale, dotato di alcune funzionalità che mancavano in molti altri concorrenti dell'epoca. Una per tutte: ad ogni modifica dei valori nel foglio, tutte le formule vengono ricalcolate in modo efficiente e preciso. Ah, e anche molto velocemente!

Si vedano le figure sotto.

P	D	E	F	G
1	Prezzo	Totale	Prov. %	Saldo
2	33000	396000	12	47520
3	50000	650000	11	71500
4	17000	425000	13	55250
5	150000	750000	8	60000
6	125000	500000	9	45000
7	68000	476000	9	42840
8	Totale:	3197000		322110

P	D	E	F	G
1	Prezzo	Totale	Prov. %	Saldo
2	33000	396000	12	47520
3	37500	487500	11	53625
4	17000	425000	13	55250
5	150000	750000	8	60000
6	125000	500000	9	45000
7	68000	476000	9	42840
8	Totale:	3034500		304235

Tasti speciali di Calc Result

F7 - Il tasto funzione F7 é la chiave di accesso a tutti i comandi e le funzioni di Calc Result.

SPACE - Abilita la differenza tra una LABEL (testo) ed un VALUE (un valore o una formula). Per inserire del testo dobbiamo prima premere SPACE e poi cominciare a scrivere il testo che vogliamo vedere apparire nella cella.

Per copiare una formula da un campo ad un altro:

- spostarsi sulla cella che contiene che la formula che





vogliamo copiare; nel nostro caso E3 contenente la formula C3*D3

- premere F7
- premere E - Edit
- premere R - Replicate

apparirà a video la funzione:

Replicate

SOURCE RANGE TARGET RANGE

FROM ... TO ... FROM ... TO ...

dato che siamo partiti dalla cella E3, la prima parte della formula é già compilata: FROM E3

- premere RETURN per conferma TO E3
 - spostarsi sulla cella E13 (vedi figura sotto)
 - premere RETURN per confermare FROM E13
 - premere nuovamente RETURN per confermare TO E13
- Dopodiché dovremo scegliere se copiare la formula in modo Absolute o Relative
- premere R per Relative
 - premere R per Relative

Dopo un veloce refresh la formula presente nella cella E3 viene copiata nella cella E13 e ricalcolata con i valori presenti di riferimento C13*D13

Attenzione: nel caso sbagliassimo a digitare una formula o a scrivere una label, possiamo svuotare la cella utilizzando la funzione: F7 + B

Esattamente come Excel, anche Calc Result dispone di una nutrita lista di funzione utilizzabili nelle formule. Si veda la figura 2 a pagina 16.

Vediamone insieme qualche esempio!

Funzione SUM

La funzione SUM, come in Excel, permette di sommare i valori specificati nel range di celle indicati dalla formula. Nel nostro caso (vedi figura 1) sommerà i valori contenuti nelle celle da E3 a E13. Per inserire questa formula basta

P	2	Marca	Q.ta'	Prezzo	Totale
	3	BMW	12	33000	396000
	4	AUDI	13	35000	455000
	5	FIAT	20	17000	340000
	6	Ferrari	5	150000	750000
	7	Porsche	4	125000	500000
	8	Jaguar	7	68000	476000
	15	Totale:			2517000

Fig. 1 - La funzione SUM

digitare SUM(E3:E13) nella cella di destinazione. Proprio come in Excel!

Funzione MEAN

La funzione MEAN permette di trovare il valore medio di una lista di valori (figura sotto).

P	2	D	E	F	G
	1	Prezzo	Totale	Prov. %	Saldo
	3	33000	396000	12	47520
	4	37500	487500	11	53625
	5	17000	425000	13	55250
	8	150000	750000	8	60000
	11	125000	500000	9	45000
	13	68000	476000	9	42840
	15	Totale:		3034500	304235

IF ... THEN ... ELSE

E' possibile anche utilizzare il costrutto IF ... THEN ... ELSE nei nostri fogli di calcolo.

Ecco un esempio di questa funzione:

IF F16 >= 10 THEN E19 ELSE E20

Si vedano le figure sottostanti per rispettivamente i valori

P	2	D	E	F	G
	1	Prezzo	Totale	Prov. %	Saldo
	3	33000	396000	8	31680
	4	37500	487500	11	53625
	5	17000	425000	13	55250
	8	150000	750000	8	60000
	11	125000	500000	9	45000
	13	68000	476000	9	42840
	15	Totale:		3034500	288395
	16	Media:		505750	48065.8
	19	Vero		% >= 10	NA
	20	Falso		1	0

P	2	D	E	F	G
	1	Prezzo	Totale	Prov. %	Saldo
	3	33000	396000	11	43560
	4	37500	487500	11	53625
	5	17000	425000	13	55250
	8	150000	750000	8	60000
	11	125000	500000	9	45000
	13	68000	476000	9	42840
	15	Totale:		3034500	300275
	16	Media:		505750	50045.8
	19	Vero		% >= 10	NA
	20	Falso		1	1

VERO e FALSO.

La stessa formula può essere riscritta:

IF F16 >= 10 THEN (1) ELSE (0)

Da notare che le formule possono essere scritte omettendo





gli spazi:

IFF16>=10THEN(1)ELSE(0)

Spazi che invece ho preferito mantenere nell'articolo per facilitarne la lettura delle formule.

Purtroppo nell'espressione non é possibile utilizzare del testo (o almeno io non sono riuscito a trovarlo), inoltre la stessa funzione che punti a delle celle contenenti del testo genera un errore NA (figura sotto).

P	D	E	F	G
2	Prezzo	Totale	Prov. %	Saldo
3	33000	396000	8	31680
4	37500	487500	11	53625
5	17000	425000	13	55250
6	150000	750000	8	60000
7	125000	500000	9	45000
8	68000	476000	9	42840
9	Totale:	3034500		288395
10	Media:	505750	9.66667	48065.8
11	% >= 10		1	0
12	Vero		1	0
13	Falso		0	0

Suddivisione dello schermo

Calc Result é dotato anche di alcune funzionalità decisamente incredibili per il 1983. Una di queste é sicuramente la possibilità di duplicare lo schermo, sia in verticale che in orizzontale.

Spostatavi all'inizio di una riga qualsiasi e:

- premete F7
- premete E - Edit
- premete S - Split
- premete H - Horizontal (orizzontale)

In questo modo il vostro foglio di calcolo sarà suddiviso in due schermi indipendenti (figura sotto).

P	A	B	C	D
2	Agente	Marca	Q.ta	Prezzo
3	Roger	BMW	12	33000
4	Franco	AUDI	13	37500
5	Lupo	FIAT	25	17000
6	Marco	Ferrari	5	150000

P	A	B	C	D
2	Agente	Marca	Q.ta	Prezzo
3	Roger	BMW	12	33000
4	Franco	AUDI	13	37500
5	Lupo	FIAT	25	17000
6	Marco	Ferrari	5	150000
7	Jimmy	Porsche	4	125000

Potrete spostarvi da uno schermo all'altro premendo F2. Da notare che, cambiando un valore in uno dei due schermi, le formule verranno istantaneamente ricalcolate su entrambi. Che dire, decisamente incredibile.

Per tornare alla visualizzazione normale vi basterá premere

F7 e poi L.

Sottofinestre

Le sorprese non si fermano alla suddivisione dello schermo. É infatti possibile creare delle sottofinestre in qualsiasi parte dello schermo.

Per provare questa funzionalità non dovrete fare altro che:

- spostarvi su una cella qualsiasi, nel nostro caso B21
- premere F7
- premere E - Edit
- premere W - Window
- confermare il punto di partenza della finestra: B21
- digitare o spostarsi nella cella che delimitará la parte in basso a destra della finestra: C25

Dopo un breve refresh una nuova finestra verrà inserita nelle coordinate indicate (si veda figura sotto).

P	A	B	C	D
7	Lupo	FIAT	25	17000
8	Marco	Ferrari	5	150000
9	Jimmy	Porsche	4	125000
10	George	Jaguar	7	68000
11				Totale:
12				Media:
13				Vero
14				
15				
16				
17				
18				
19				
20				
21	Agente	Marca	Q.ta	
22				
23	Roger	BMW	12	
24				
25	Franco	AUDI	13	
26				

Grafici

Ovviamente non poteva mancare la possibilità di generare grafici. Purtroppo devo ammettere che questa funzionalità, seppur rapida e visivamente accattivante, é decisamente limitata. É possibile infatti creare grafici (istogrammi) solo dei valori contenuti in una determinata riga o colonna. Nel nostro caso abbiamo rappresentato i valori della colonna D.

Spostatavi nella colonna D e:

- premete F7
- premete E - Edit
- premete G - Graphic
- premete C - Column

Digitate i valori minimi e massimi della scala di riferimento: nel nostro caso 10000 e 160000

Dopodiché digitate il titolo del vostro grafico: Prezzi auto. Il risultato é visibile in figura 3.

In conclusione

Mi sono divertito molto a provare Calc Result perché é un software decisamente all'avanguardia per il periodo in cui é stato concepito. Questo foglio di calcolo dimostra una volta di piú che il Commodore 64 era decisamente



**GROUP 1**

COUNT Gives the number of co-ordinates containing a constant or valid formula (without NA or error).

MAX Chooses the largest value within a given area

MIN Chooses the smallest value within a given area

MEAN Gives average value for a certain area

STDDEV Calculates the standard deviation of a given area

SUM Calculates the sum of a given area

Structure: FUNCTION (argument, argument, argument.....)

Example: MEAN(A1:A5,B1:B5)

GROUP 2

NPV To calculate the net present value

Structure: FUNCTION (per cent, argument: argument)

Example: NPV(.10,C1:G15)

GROUP 3

NA For a co-ordinate which is missing a value or a formula which refers to an invalid co-ordinate

PI Gives the constant π value

Structure: FUNCTION

Example: NA

GROUP 4

ABS, ARCCOS, ARCSIN, COS, EXP, FRAC, INT, LN, LOG10, RND, SIN, SQRT and TAN

Structure: FUNCTION (expression)

Example: SIN(A1+1*10/B7)

GROUP 5

IF THEN ELSE

Structure: IF expression THEN expression ELSE expression

Example: IF A1=10 THEN B1+2 ELSE 5

GROUP 6

OR, AND and NOT

Structure: FUNCTION within IF THEN ELSE

Example: IF A1=10 OR C1=5 THEN B1+2 ELSE 5

Fig. 2 - Funzioni matematiche di Calc Result

un computer dalle mille potenzialità: strumento di gioco per milioni di ragazzini ma anche strumento di lavoro per chi aveva la necessità di utilizzarlo in ambito office.

Se poi pensiamo che tutte le cose che vi ho descritto funzionano con 64K di RAM ed un processore ad 1MHz, beh, c'è soltanto da fare i complimenti ai programmatori.

Decisamente incredibile!

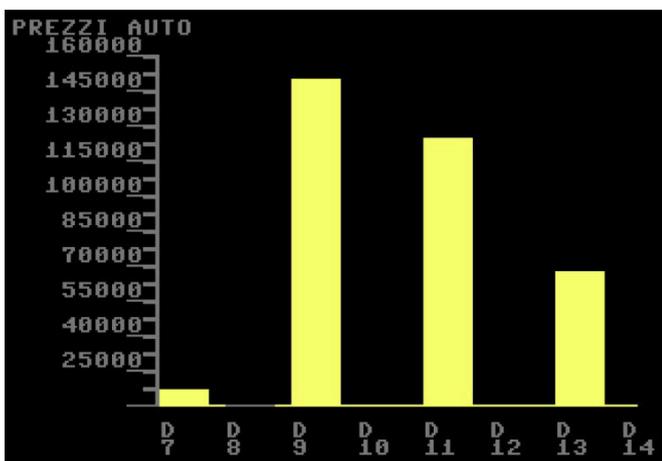


Fig. 3 - I grafici di Calc Result

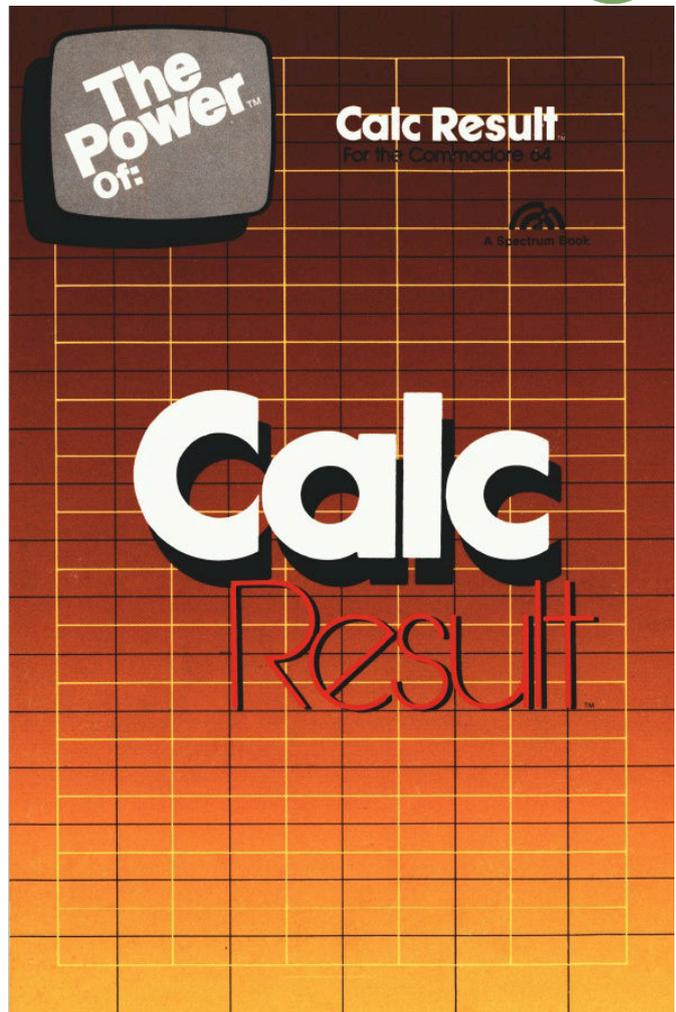


Fig. 4 - Il libro The Power of Calc Result

Per approfondire:

- Per chi volesse approfondire lo studio di **Calc Result**, lascio il link al suo **manuale**, veramente ben fatto e dettagliato:

<https://commodore.software/downloads/download/211-application-manuals/13357-calc-result>

- E' inoltre possibile consultare il libro **The Power of Calc Result**, contenente una serie di esercizi utili a comprendere il funzionamento di questo spreadsheet e come poteva essere utilizzato in ambito Office negli anni '80:

https://commodore.bombjack.org/books/commodore/books/The_Power_of_Calc_Result_for_the_Commodore_64.zip

- **Calc Result** invece può essere scaricato da:

<https://edgeemu.net/details-2855.htm>





Un esperimento in Assembly per il C64

di Eugenio Rapella

Nel mio vecchio libro Elementi di base di Probabilità e Statistica avevo presentato un programma, OPINIONE, che avevo realizzato partendo da una proposta apparsa nella rubrica Ricreazioni al calcolatore del numero di giugno 1985 di Le Scienze, la versione italiana di Scientific American.

Rivedendo quel capitolo, ho pensato questa variazione dove ho cambiato un po' le carte in tavola e ho eliminato l'aspetto probabilistico.

All'inizio del programma, lo schermo del Commodore 64 viene casualmente riempito da due simboli "*" e "." che rappresentano due stati, in qualche modo, contrastanti o "in conflitto" (potrebbero essere conservatori/riformisti; interisti/milanisti, cellule cutanee normali/cellule alterate). Il simbolo viene scelto casualmente (con probabilità del 50% "*", con probabilità 50% ".") per ciascuna delle 1000 locazioni presenti nello schermo del C64 (sono 25 righe per 40 colonne). Si comincia dunque da una situazione in cui i due simboli sono ben mescolati, sia come consistenza numerica, sia come disposizione geografica.

A questo punto si apre il ciclo principale: si sceglie a caso un elemento della griglia, chiamiamolo alfa (vengono esclusi quelli "sul bordo", ovvero quelli della prima e dell'ultima riga, della prima e dell'ultima colonna) e si contano quanti asterischi sono presenti tra gli otto "vicini", ovvero quelli della locazione subito a nord, a sud, a est, a ovest, a nord-est, a nord-ovest, a sud-est e a sud-ovest. Se il numero di asterischi è maggiore di 4, il simbolo in alfa si trasforma in un "asterisco" (indipendentemente dal suo contenuto originale), se è minore di 4 il simbolo in alfa diviene un "punto" mentre se è esattamente 4 il contenuto di alfa rimane inalterato (chiamiamola "regola del quattro"). In altre parole, la situazione del vicinato influenza alfa e fa sì che alfa assuma la "posizione" della maggioranza dei suoi vicini (per gli alfa sono state escluse le locazioni del bordo in modo che ogni alfa abbia tutti gli otto "vicini"; un altro modo per risolvere la questione, e che avevo adottato nel programma OPINIONE, è quello di "saldare" l'ultima riga con la prima, ottenendo un cilindro, e di collegare poi, piegando il cilindro, il cerchio di sinistra con quello di destra. Si ottiene così quello che in geometria è chiamato "toro" – una ciambella, un

salvagente – e nessuna delle caselle sarebbe stata sul bordo). Una volta eseguita la trasformazione, il programma riparte con la scelta casuale di un nuovo alfa e una nuova, eventuale, trasformazione. Cosa succede a lungo andare? È chiaro che zone dove tutti sono della stessa opinione, rimangono stabili perché gli alfa pescati lì dentro mantengono il loro stato. Nelle zone "di confine" si presentano configurazioni che, in qualche modo, si... stabilizzano.

Ad esempio, una configurazione come quella che segue risulta "stabile": contando i vicini delle varie caselle, si nota che non ci saranno cambiamenti secondo la "regola del quattro":

.
.
.	.	.	*	*	*
.	.	*	*	*	*	*	.	.	.
.	.	*	*	*	*
.	.	.	*	*	*
.
.

Nel programma OPINIONE che avevo presentato nel libro, il gioco era diverso: venivano pescati due simboli contigui con opinioni differenti e ciascuno dei due cercava di convincere l'altro a cambiare bandiera; i vicini della coppia, in numero di dieci, cercavano di influenzare i due e la probabilità di successo di uno o dell'altro era legata anche a una probabilità P, richiesta all'inizio del programma, che forniva il "peso" del vicinato. In OPINIONE si andava verso il totalitarismo nel senso che, a lungo andare, la griglia si riempiva totalmente di puntini o di asterischi (l'aspetto più interessante del gioco era costituito dalle configurazioni intermedie e dal tempo necessario, inteso come numero di cicli, per raggiungere l'uniformità: tanto più i vicini erano persuasivi, tanto prima si raggiungeva l'uniformità di ... opinione).

Nella nostra nuova versione del gioco, lasciamo che sia il C64 a mostrarci quello che succede con l'andar del tempo. Ecco una possibile soluzione in Basic:





```

5 w=rnd(-ti)
10 poke 53280,0 : rem bordo nero
20 poke 53281,0 : rem sfondo nero
30 for t=0 to 999
40 hh=46 : rem puntino
50 if rnd(0)<.5 then hh=42 : rem asterisco
60 poke 1024+t,hh
70 poke 55296+t,1: rem colore bianco dei
  caratteri
80 next t
90 ns=0
100 x=int(rnd(1)*38)+1
110 y=int(rnd(1)*23)+1
120 n=1024+x+40*y
130 if(peek(n-40)=42) then ns=ns+1
140 if(peek(n+40)=42) then ns=ns+1
150 if(peek(n-1)=42) then ns=ns+1
160 if(peek(n+1)=42) then ns=ns+1
170 if(peek(n-41)=42) then ns=ns+1
180 if(peek(n-39)=42) then ns=ns+1
190 if(peek(n+39)=42) then ns=ns+1
200 if(peek(n+41)=42) then ns=ns+1
210 if ns>4 then poke n,42
220 if ns<4 then poke n,46
230 goto 90

```

Il ciclo FOR-NEXT dalla 30 alla 80 passa in rassegna le 1000 caselle dello schermo, la locazione 1024 corrisponde all'angolo in alto a sinistra e viene riempita con un "punto" (poke....., 46) o con un "asterisco" (poke, 42), con probabilità $\frac{1}{2}$, nell'istruzione 60 quando $t=0$, poi tocca alla 1025, $t=1$, e così via. Alla 70 si approfitta del ciclo per inserire il "bianco" come colore associato alla locazione (nel C64, la mappa dei colori dei caratteri dello schermo inizia alla locazione 55296).

Dopo questa operazione preliminare, inizia il programma vero e proprio: istruzioni dalla 90 alla 230. ns è la variabile che contiene il numero di asterischi del vicinato; inizialmente viene posta uguale a zero (istr. 90). La 100 origina un numero casuale che sarà il numero della colonna della locazione scelta per il gioco, la nostra alfa. Le colonne sono 40 e sono numerate da 0 a 39. Poiché vogliamo evitare la prima e l'ultima, desideriamo che x sia da 1 a 38. La funzione $\text{rnd}(1)$ fornisce un numero decimale, casuale, da 0 a 1: 0 compreso, 1 escluso: $0 \leq \text{rnd}(1) < 1$.

Moltiplicando per 38, si ottiene un numero decimale tra 0 e 38 con 38 escluso; prendendone la parte intera $\text{int}(\text{rnd}(1)*38)$ si ha un numero intero da 0 a 37; con il "+ 1" finale si arriva infine al valore casuale da 1 a 38

(istr. 100). Discorso del tutto analogo per l'istruzione 110 che fornisce un valore (y) casuale da 1 a 23 che consente di determinare la riga di alfa escludendo la prima e l'ultima (come si è detto, le righe sono 25 numerate da 0 a 24).

Con l'istruzione 120 viene calcolato il numero della locazione di memoria dello schermo del C64 corrispondente alla riga x e alla colonna y , numero che viene assegnato alla variabile n .

Le otto istruzioni dalla 130 alla 200 operano tutte allo stesso modo: ciascuna controlla il contenuto della locazione di uno degli 8 "vicini", se si tratta di un asterisco la variabile ns viene incrementata di una unità in modo che, quando si arriva all'istruzione 210, ns contiene il numero complessivo degli asterischi dei vicini di alfa, un numero intero tra 0 e 8.

Siamo alla fine: istruzione 210, se ns supera 4, nella locazione di alfa, n , viene inserito un asterisco (indipendentemente da ciò che c'era in quel momento); se invece, istruzione 220, il numero di asterischi dei vicini è inferiore a 4 (il che significa che i puntini saranno da 5 a 8), in n va un puntino (se $ns=4$, il contenuto di n rimane inalterato). A questo punto si ritorna alla istr. 90 per la scelta di un nuovo alfa. Il programma non ha termine (andrà interrotto manualmente) e sullo schermo del C64 potremo assistere all'evolversi della situazione. I simboli che compaiono sul bordo non saranno mai modificati, ma influenzeranno i loro vicini...

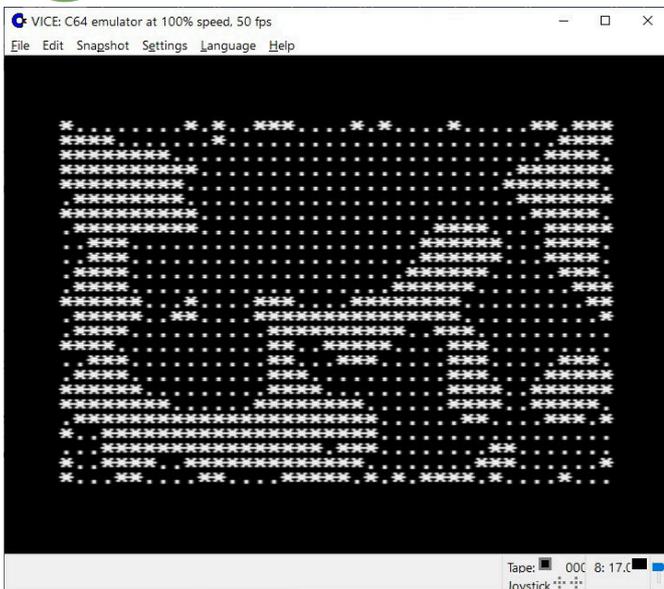
Il programma Basic è molto lento, già la creazione della configurazione iniziale porta via parecchi secondi.

Ecco come si presenta lo schermo dopo la fase d'inizio:

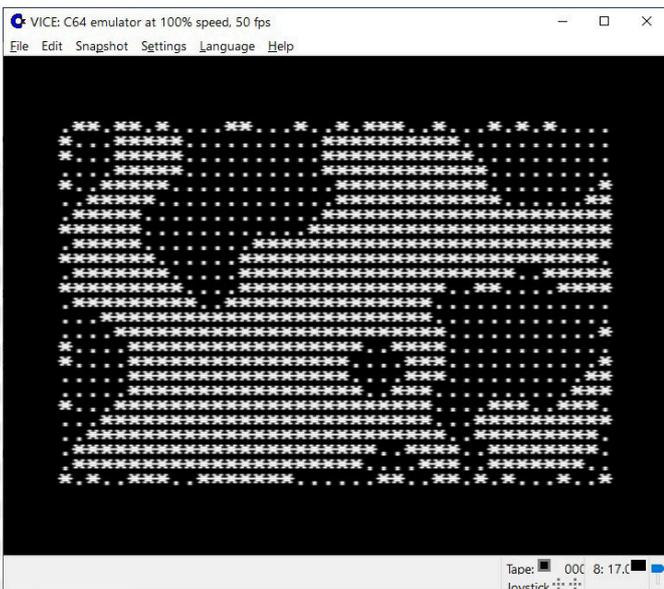
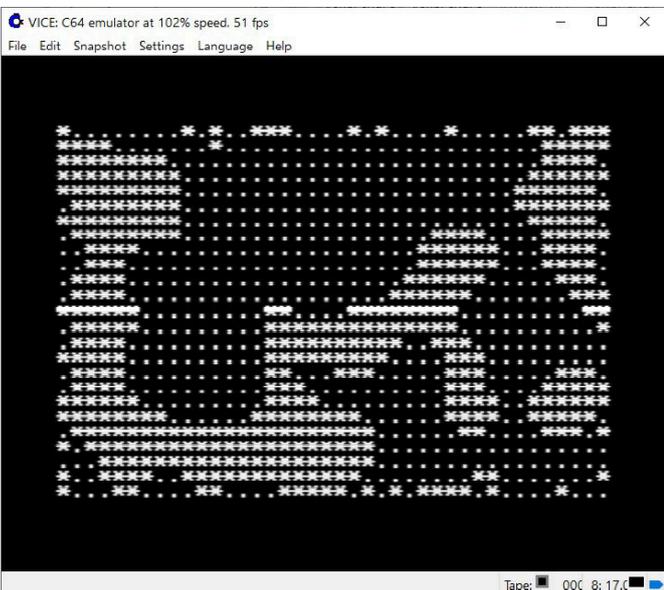


Ci vogliono circa 5 minuti per ottenere questa configurazione:

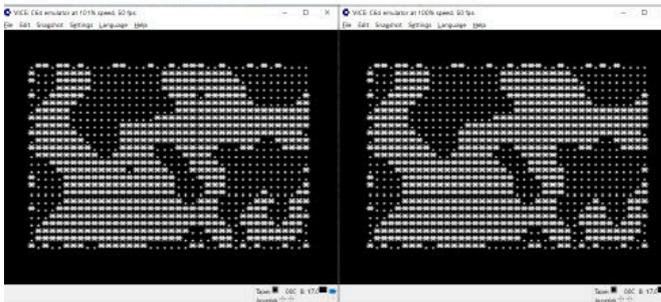
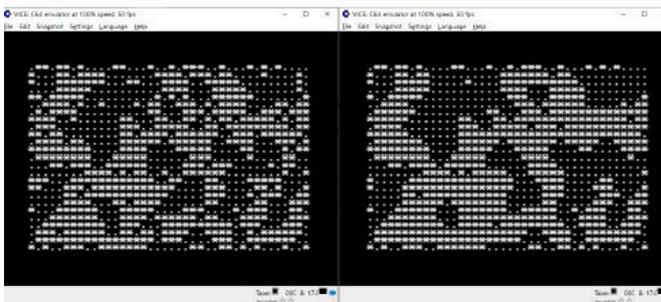
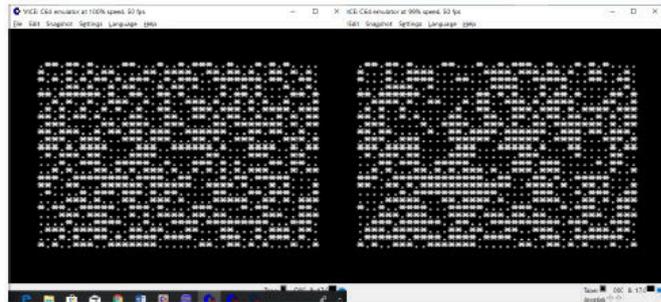




e altri 5 circa per ottenere questa configurazione che sembra essere abbastanza stabile:



La configurazione "finale", che dipende da quella iniziale e dai simboli lungo il bordo - quelli che non cambiano -, non è facilmente prevedibile; la parte interessante e



divertente (per me lo è) del programma consiste proprio nell'assistere all'evoluzione del disegno fino al pattern finale. Perché l'elaborazione è così lenta? Beh, l'interprete Basic del nostro C64 non è certo un campione di velocità, inoltre, se all'inizio assistiamo a frequenti cambiamenti di simboli, con l'andar del tempo, tutto sembra rallentare...

Il motivo è semplice: si creano zone stabili di area sempre maggiore e, poiché il candidato alfa viene sempre scelto a caso, il più delle volte capita all'interno di queste zone. Il Commodore lavora sempre alla stessa velocità, ma in queste zone il simbolo in alfa viene riconfermato e, di fatto, nulla cambia sullo schermo.

Ho provato a compilare il codice Basic (con Blitz Compiler, come avevo fatto, ai tempi, con OPINIONE); naturalmente, il codice compilato è un bel po' più veloce, ma... Ho voluto fare un piccolo esperimento con l'Assembly del 6502, utilizzando CBM Prg Studio (che ho utilizzato anche per il codice Basic e, naturalmente, come si vede dalle immagini, tutto si conclude sul mitico emulatore VICE).

Il codice che ho prodotto è senz'altro molto, molto migliorabile (si tratta, appunto, di un esperimento) ma, come prevedibile, il confronto in termini di velocità è decisamente a favore di quest'ultimo!

Il codice che ho prodotto è abbastanza ricco di commenti;





è servito a chiarirmi alcune idee, anche se un po' di dubbi sono rimasti.

Eccolo qui:

```

*=$C000; sys 49152
punto=$2e; punto è #46
asterisco=$2A ; asterisco #42
inizioSchermo=$0400 ; 1024 in base 10
inizioColore=$D800; 55296 in base 10
    LDA #0 ; colore NERO
    STA $D020 ; bordo nero
    STA $D021 ; sfondo nero
;Questa Routine (ripresa da Internet) genera un
numero casuale da 0 a 255
;nella locazione $D41B. Richiede una
inizializzazione ... una tantum:
;***inizializza RND ***
    LDA #$FF ; maximum frequency value
    STA $D40E ; voice 3 frequency low byte
    STA $D40F ; voice 3 frequency high byte
    LDA #$80 ; noise waveform, gate bit off
    STA $D412 ; voice 3 control register
;*****
;***** INIZIALIZZAZIONE SCHERMO *****
; Poichè lo schermo contiene 1000 caratteri,
viene riempito in quattro
; tornate di 250 caratteri ciascuna. Il contatore
è il registro X
; che parte da 250 e si decrementa (dex) fino a zero
; *** i=0 ***
    ldx #250
loop
    lda #asterisco
    sta $0346 ; locazione che contiene il simbolo
da stampare
    lda $D41B ; carica in A un RND da 0 a 255
    cmp #$80 ; confronta RND con 128 (a metà tra
0 e 255)
    bcs labb ; se A > 128 va a LABB
    lda #punto ; se A <= 128 "*" viene sostituito
da "."
    sta $0346 ; la locazione contiene "." o "*"
con prob 1/2
labb
    lda $0346
    sta $0400-1,x ; memorizza il simbolo sullo schermo
    lda #$1 ; colore bianco per i caratteri
    sta inizioColore-1,x
    dex
    bne loop ; se x non è zero torna sopra
; *** i=1 ***

```

```

    ldx #250
loop1
    lda #asterisco
    sta $0346 ; locazione che contiene il simbolo
da stampare
    lda $D41B ; carica in A un RND da 0 a 255
    cmp #$80 ; confronta RND con 128 ecc.
    bcs labb1
    lda #punto
    sta $0346
labb1
    lda $0346
    sta $0400+249,x
    lda #$1
    sta inizioColore+249,x
    dex
    bne loop1
; *** i=2 ***
    ldx #250
loop2
    lda #asterisco
    sta $0346 ; locazione che contiene il simbolo
da stampare
    lda $D41B ; carica in A un RND da 0 a 255
    cmp #$80 ; confronta RND con 128 ecc
    bcs labb2
    lda #punto
    sta $0346
labb2
    lda $0346
    sta $0400+499,x
    lda #$1
    sta inizioColore+499,x
    dex
    bne loop2
; *** i=3 ***
    ldx #250
loop3
    lda #asterisco
    sta $0346 ; locazione che contiene il simbolo
da stampare
    lda $D41B ; carica in A un RND da 0 a 255
    cmp #$80 ; confronta RND con 128
    bcs labb3
    lda #punto
    sta $0346
labb3
    lda $0346
    sta $0400+749,x
    lda #$1
    sta inizioColore+749,x

```





```

dex
bne loop3
;
; ***** FINE INIZIALIZZAZIONE SCHERMO *****
zpl0=$fb ; locazioni di pagina zero
zpl1=$fc ; per l'indirizzamento indiretto
main
; Creiamo x da 0 a 37
alfa5
lda $d41b ; RND 0 - 255 numero casuale in A
(Accumulatore)
and #$3f ; ora RND è tra 0 - 63
cmp #$26 ; confronta con 38
bcs alfa5 ; torna ad alfa5 se A>=38
sta $0342 ; questa locazione contiene x
; Creiamo y da 0 a 22
alfa6
lda $d41b ; RND 0 - 255
and #$1f ; RND 0 - 31
cmp #$17 ; confronta con 23
bcs alfa6 ; torna ad alfa6 se A>=23
sta $0343 ; questa locazione contiene y
sta $0344 ; copia di y
asl $0344
asl $0344 ; <- 4y
lda $0343 ; y -> A
adc $0344 ; 5y -> A
sta $0344 ; <- 5y (1 byte, max 115)
asl $0344 ; <- 10y (1 byte, max 230)
clc
lda #$0
sta $0345
asl $0344 ; <- 20y
rol $0345 ; per via del riporto
clc
asl $0344 ; <- 40y
rol $0345 ; per via del riporto
clc
lda $0344 ; low di 40y -> A
adc $0342 ; + x
sta $0344
lda $0345
adc #$0
sta $0345
clc
lda $0344
adc #$00 ; low byte di inizioSchermo
sta $0344 ; low di N
lda $0345
adc #$04 ; high byte di inizioSchermo
sta $0345 ; high di N

```

```

; ora N contiene 1024+x+40*y, non è il nostro
"alfa", ma
; la locazione una riga in meno e una colonna in meno
; (l'angolo superiore sinistro, rispetto ad alfa).
; In altre parole, anziché scegliere a caso alfa
e contare
; il numero di asterischi "attorno" ad alfa,
conviene scegliere
; a caso l'angolo superiore sinistro del riquadro
(beta) e contare
; gli asterischi presenti in beta, beta+1, beta+2,
beta+40, beta+42
; (beta+41 corrisponde ad alfa) e infine beta+80,
beta+81 e beta+82;
; in questo modo si eseguono solo addizioni.
; Copiamo N (beta) nella locazione utilizzata
successivamente:
lda $0344
sta $033e
lda $0345
sta $033f
; Conteggio del numero di asterischi: valore
aggiornato nel registro X
ldx #$0
; $033e (low byte) e $033f (high byte) contengono
la locazione dello schermo
; oggetto del controllo (contiene un asterisco?).
All'inizio è "beta", poi
; viene aggiornata sommando 1 (per ottenere
beta+1), poi sommando un altro 1
; (per ottenere beta+2), poi si addiziona 38 (per
ottenere beta+40), poi si
; addiziona 2 (per ottenere beta+42), poi gli
aggiornamenti proseguono con +38,
; +1 e infine +1. La parte di codice tra "++++"
e "-----" viene dunque
; ripetuta 8 volte: all'inizio l'INCREMENTO viene
caricato in A e la subroutine
; "gamma" si incarica di aggiornare $033e e $033f.
; +++++ (inizio: siamo a NW di alfa, angolo
superiore sinistro rispetto alfa)
; N,S,E,W = nord, sud, est, ovest, NE = nord-est ecc.
lda $033e
sta zpl0
lda $033f
sta zpl1
ldy #$00
lda (zpl0),y
cmp #asterisco
bne alfa0
inx

```





```

alfa0
; -----
    lda #$01 ; INCREMENTO +1 (siamo a N di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11
    ldy #$00
    lda (zp10),y
    cmp #asterisco
    bne alfa1
    inx
alfa1
; -----
    lda #$01 ; INCREMENTO +1 (siamo a NE di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11
    ldy #$00
    lda (zp10),y
    cmp #asterisco
    bne alfa2
    inx
alfa2
; -----
    lda #$26 ; INCREMENTO +38 (siamo a W di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11
    ldy #$00
    lda (zp10),y
    cmp #asterisco
    bne alfa40
    inx
alfa40
; -----
    lda #$02 ; INCREMENTO +2 (siamo a E di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11

                                ldy #$00
                                lda (zp10),y
                                cmp #asterisco
                                bne alfa42
                                inx
alfa42
; -----
    lda #$26 ; INCREMENTO +38 (siamo a SW di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11
    ldy #$00
    lda (zp10),y
    cmp #asterisco
    bne alfa80
    inx
alfa80
; -----
    lda #$1 ; INCREMENTO +1 (siamo a S di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11
    ldy #$00
    lda (zp10),y
    cmp #asterisco
    bne alfa81
    inx
alfa81
; -----
    lda #$1 ; INCREMENTO +1 (siamo a SE di alfa)
    jsr gamma
; +++++
    lda $033e
    sta zp10
    lda $033f
    sta zp11
    ldy #$00
    lda (zp10),y
    cmp #asterisco
    bne alfa82
    inx
alfa82
; -----
; Ora il registro X contiene un intero da 0 a 8
; pari al numero

```





```
; di asterischi che circondano la locazione "alfa".
; se X<4 in "alfa" va un puntino; se X>4 va un
asterisco.
```

```
; La locazione "beta" era conservata in $0344
(low) e $0345 (high),
; copiamo i valori in $033e e $033f in modo di
sfruttare la subroutine
; "gamma" per aggiungere 41 e ottenere la
locazione "alfa"
```

```
lda $0344
sta $033e
lda $0345
sta $033f
lda #$29 ; corrisponde a 41 in base 10
jsr gamma
cpx #$4
bcs alfabis ; salta se X >= 4
lda $033e
sta zp10
lda $033f
sta zp11
ldy #$0
lda #punto ; (dunque, se X<4 in "alfa" va un
```

puntino)

```
sta (zp10),y
jmp out ; torniamo all'inizio
```

alfabis

```
cpx #$4
beq out ; salta se X=4
```

```
; se siamo qui significa che X>4 e in "alfa" va
un asterisco:
```

```
lda $033e
sta zp10
lda $033f
sta zp11
ldy #$0
lda #asterisco
sta (zp10),y
out
jmp main
rts
```

```
; subroutine gamma: somma il contenuto di A al
contenuto
```

```
; di $033e (low byte) e di $033f (high byte)
```

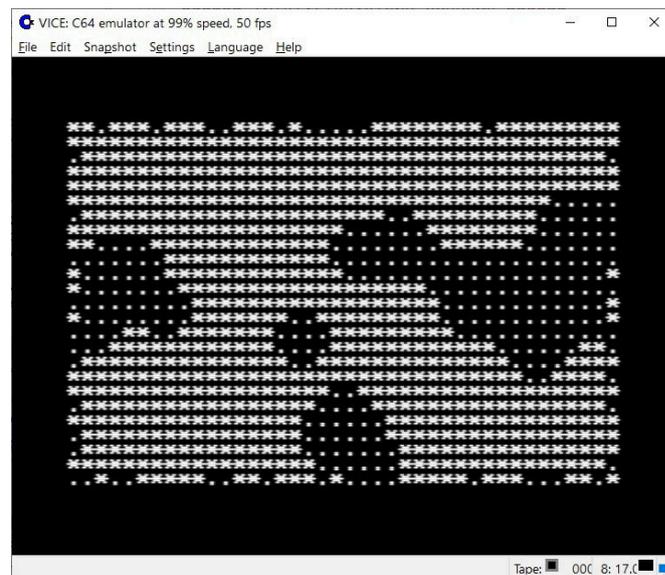
```
gamma
clc
adc $033e
sta $033e
bcc teta
lda #$01
clc
```

```
adc $033f
sta $033f
teta
rts
; fine subroutine gamma
```

Il gioco/programma si presta a qualche variazione. Una possibilità è quella di privilegiare uno dei due simboli nella configurazione iniziale. Se, ad esempio, facciamo sì che, all'inizio, la probabilità di "*" sia del 75% contro un 25% per ".":

```
loop
lda #asterisco
sta $0346 ; locazione che contiene il simbolo
da stampare
lda $D41B ; carica in A un RND da 0 a 255
cmp #$40 ; confronta RND con 64 (a 1/4 tra 0 e 255)
bcc labb ; se A > 64 va a LABB
lda #punto ; se A <= 64 "*" viene sostituito da "."
sta $0346 ; la locazione contiene "." con prob 1/4
labb
```

anche la configurazione finale conterrà molti più asterischi di puntini:



È anche possibile modificare la "regola del quattro" in modo da privilegiare uno dei due simboli.

Un'altra possibilità è quella di mantenere la "regola del quattro", ma di farla lavorare ... al contrario: se nel vicinato il numero X di asterischi è maggiore di quattro, la casella al centro si trasforma in un punto; se $X < 4$ la casella al centro diviene un asterisco. Quanti più vicini sono di un tipo, tanto più il soggetto al centro passa dall'altra parte (un comportamento analogo al programma ANTIVOTERS che avevo preso in considerazione nel libro).

Come evolve la configurazione iniziale con questa regola da "bastian contrario"? Non c'è che da provare, anche se possiamo immaginare che verranno evitate concentrazioni





dello stesso simbolo (una disposizione dei due simboli "a scacchiera" risulterebbe stabile).

Le variazioni da apportare al codice Assembly sono minime, basta scambiare "punto" e "asterisco" nel codice dopo il conteggio del numero degli asterischi (nel programma in Basic, basterà scambiare tra loro il 42 il 46 nelle istruzioni 210 e 220):

```

cpx #$4
  bcs alfabis ; salta se X >= 4
  lda $033e
  sta zp10
  lda $033f
  sta zp11
  ldy #$0
  lda #asterisco
  sta (zp10),y
  jmp out
alfabis
  cpx #$4
  beq out
  lda $033e
  sta zp10

```

```

lda $033f
sta zp11
ldy #$0
lda #punto
sta (zp10),y

```

out

Senza variare i nomi delle costanti, si possono associare a "punto" e "asterisco" codici ASCII diversi in modo da ottenere effetti grafici più ... plateali. All'inizio del codice Assembly modifichiamo le due righe:

```
punto=$2e; punto è #46
```

```
asterisco=$2A ; asterisco #42
```

in

```
punto=$20; ora il codice è 32: lo spazio, quadrato
"vuoto"
```

```
asterisco=$A0 ; qui il codice ASCII è 160: quadrato
"pieno"
```

(nel codice Basic si sostituirà 46 con 32 e 42 con 160 nelle istruzioni 40, 50 e dalla 130 alla 220) e lanciamo il programma in Assembly con la variazione ... al contrario.

Ecco qualche risultato con i nuovi e vecchi simboli:





Grafica... Che passione!

Parte IV - Il formato grafico KLA e come interagirvi

di Gianluca Girelli

Come si sa, l'appetito vien mangiando, ed eccoci quindi di nuovo su queste pagine a parlare di grafica per i sistemi ad 8 bit, primariamente C64 e C128. Dopo aver imparato sul Numero 40 come sono strutturati gli assetti che contribuiscono a formare un'immagine multicolor bitmap su alcune delle vecchie macchine Commodore, vediamo ora come importare tali dati in un programma di alto livello per rieditare l'immagine; sarà così più semplice ed efficace preparare degli assetti migliori per i nostri progetti futuri.

Come già detto in precedenza, esistono oggi strumenti per la creazione e la manipolazione di immagini grafiche estremamente potenti che, mimando alla perfezione la meccanica di gestione delle bitmap secondo le specifiche Commodore, permettono di realizzare sul proprio PC set di caratteri ridefiniti, sprites e figure in maniera semplice, intuitiva e con un approccio moderno. Uno di questi strumenti è Multipaint (Figg. 1 e 2) e, attraverso di esso, è possibile creare, esportare e reimportare immagini per la maggior parte dei retrocomputers emulandone le limitazioni di colore di cui abbiamo trattato nel recente passato (ad esempio, il numero massimo di colori all'interno di una matrice di 8x8 pixels). Al momento attuale, Multipaint supporta i seguenti formati: Commodore 64 ad alta risoluzione, Commodore 64 multicolor (Fig. 3), Commodore Plus/4 Hires, Commodore Plus/4 multicolor, ZX Spectrum, ZX ULaplus, MSX 1 (Fig. 4), Amstrad CPC mode 0/1 (e 0 con Overscan), Sinclair QL e Atari ST (Fig. 5) a bassa risoluzione. (Figg. 3 - 5) Il C128 non è direttamente

supportato, ma sul precitato Numero 40 abbiamo imparato che i file che compongono le bitmap sono assolutamente compatibili tra C64 e C128 a patto di cambiare l'indirizzo di caricamento di "screen.dat".

Dopo aver risolto il problema di come venivano gestiti i file che compongono la nostra immagine multicolor ed in che parte della memoria del sistema vengono memorizzati, mi sono quindi chiesto come poter importare questi file all'interno di Multipaint. Il motivo principale è dovuto al fatto che non c'è modo di compensare totalmente la differenza nel numero di colori di una qualunque immagine presa da internet (o dalla vostra macchina fotografica) e la limitata palette (solo 16 colori) dei nostri sistemi di riferimento. Per questo motivo, nell'articolo sul num. 14, il buon Marco esortava a scegliere immagini con un numero modesto di colori; in caso contrario, il risultato del processo di "trasformazione" darà luogo ad un prodotto dalla grafica decisamente sgranata, come vedremo in seguito.

Studiando la documentazione di Multipaint ho appurato che in modalità "C64 multicolor" il programma esporta ed importa files in formato ".kla"; tale formato è quello usato da un famosissimo programma degli anni '80, chiamato "Koala Painter", che permetteva di disegnare tramite una tavoletta grafica e relativa penna ottica (Figg. 6 e 7). Sviluppato nel 1983 dalla Koala Technologies Corporation ed inizialmente pensato per un utilizzo scolastico, fu prodotto in varie versioni per l'Apple II, il TRS-80 Color Computer, gli Atari a 8 bit, il Commodore



Fig. 1 - Multipaint main screen

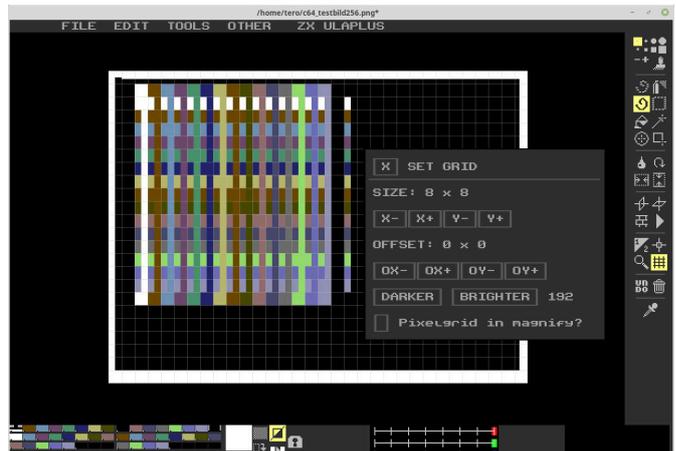


Fig. 2 - Multipaint options screen





Fig. 3 - C64 screen by davide bucci - queens

64 (ovviamente) ed anche per i PC IBM.

In particolare, la versione Commodore 64 di KoalaPainter utilizzava un formato di file abbastanza semplice che corrisponde direttamente al modo in cui la grafica bitmap viene gestita sul computer: un indirizzo di caricamento a due byte, seguito immediatamente da 8.000 byte di dati bitmap, 1.000 byte di "Video Matrix", 1.000 byte di dati "Color RAM" e di un byte per il colore di sfondo. Poichè non si tratta che di un semplice file binario di 10.003 bytes, ho pensato quindi di scrivere un programma che permettesse di estrarre i dati di interesse da un singolo file ".kla" o, viceversa, di riassemblearne uno a partire dai file ".dat" generati dal programma di Marco. Per lo scopo ho usato il linguaggio "Hollywood", un M.A.L. (Multimedia Application Layer) basato sul linguaggio LUA ed inizialmente sviluppato su Amiga di cui ho già trattato in precedenti articoli.

Nel primo caso (estrazione dati) il procedimento è il seguente:

- aprire il file .kla in lettura
- copiare il file dentro una stringa binaria
- aprire hires.dat in scrittura



Fig. 5 - Atari ST screen by Stefan Vogt_Rabenstein



Fig. 4 - MSX screen by drterrorz

- scrivere nel file l'indirizzo \$2000 (indirizzo standard)
- scrivere in hires.dat gli 8.000 byte di struttura dati a partire dal terzo (i primi due sono l'indirizzo di caricamento del .kla), poi chiudere il file
- aprire screen.dat in scrittura
- scrivere nel file l'indirizzo \$1c00 (indirizzo standard per il C128) o \$0400 (per il C64)
- scrivere in screen.dat i 1.000 byte successivi e chiudere il file
- aprire colors.dat in scrittura
- scrivere nel file l'indirizzo \$d800 (indirizzo standard)
- scrivere in colors.dat i 1.000 byte successivi e chiudere il file
- infine chiudere .kla

Si noti che in realtà manca all'appello un ultimo byte, corrispondente a quello contenente il colore di background, uguale per tutte le celle. Potete impostarlo in seguito come volete con un comando POKE o COLOR (solo per il C128).

Si noti inoltre l'uso delle stringhe per lavorare sui file binari: dal momento che in Hollywood non esiste un limite al numero massimo di caratteri di una stringa, questo approccio permette un approccio più flessibile nel copiare/incollare gruppi di dati. Si noti infine che tutti gli indirizzi vanno inseriti in formato little-endian (Nota [1]).



Fig. 6 - C64 Koala Painter





Fig. 7 - KoalaPad

Riassemblare un file ".kla" dalla serie di ".dat" è ancora più semplice ed è sufficiente:

- creare/aprire in scrittura il file .kla
- scrivervi l'indirizzo \$6000 (indirizzo standard)
- copiarvi, in sequenza, hires.dat, screen.dat e color.dat omettendo per ogni file i primi due byte
- aggiungere un byte con valore \$00 (sfondo nero)
- chiudere tutti i file.

Riporto di seguito il codice sorgente dei due programmi per un eventuale studio approfondito segnalando che, essendo Hollywood un linguaggio proprietario, è necessario acquistare il framework per poterli utilizzare ed modificare. Una versione precompilata per uso libero verrà resa disponibile tramite il sito di Retromagazine World non appena possibile.

```

;kla_extraction code
;by Gianluca Girelli 2022

;get kla file size
size = FileSize("test1.kla")
OpenFile(1, "test1.kla")

;copy binary data into alphanumeric string
data$ = ReadBytes(1, size)
;extract raw binary data from string to form

```

```

dedicates files
hires$=MidStr(data$,2,8000, #ENCODING_RAW)
screen$=MidStr(data$,8002,1000,
#ENCODING_RAW)
colors$=MidStr(data$,9002,1000,
#ENCODING_RAW)

```

```

;open first file in writing mode, add loading
address, copy raw data, close file
;addresses are added in little-endian mode
OpenFile(2, "hires.dat", #MODE_WRITE)
WriteByte(2,ToNumber("00",16))
WriteByte(2,ToNumber("20",16))
WriteBytes(2, hires$, 8000)
CloseFile(2)

```

```

;open second file in writing mode, add
loading address, copy raw data, close file
OpenFile(3, "screen.dat", #MODE_WRITE)
WriteByte(3,ToNumber("00",16))
WriteByte(3,ToNumber("1c",16))
WriteBytes(3, screen$, 1000)
CloseFile(3)

```

```

;open third file in writing mode, add loading
address, copy raw data, close file
OpenFile(4, "colors.dat", #MODE_WRITE)
WriteByte(4,ToNumber("00",16))
WriteByte(4,ToNumber("d8",16))
WriteBytes(4, colors$, 1000)
CloseFile(4)
CloseFile(1)

```

```

;kla_inception code
;by Gianluca Girelli 2022

```

```

;get single file size (actually useless,
since they are already known - 80002, 1002,
1002)
size_hir = FileSize("hires.dat")
size_scr = FileSize("screen.dat")
size_col = FileSize("colors.dat")

```

```

;open/create .kla file ...
;then add loading address in the first 2 bytes
OpenFile(1, "test2.kla", #MODE_WRITE)
WriteByte(1,ToNumber("00",16))
WriteByte(1,ToNumber("60",16))

```

```

;open hires.dat and add it to .kla, after
removing the initial address

```





Fig. 8 - Babil junior



Fig. 9 - Babil junior su Commodore 64

```

OpenFile(2, "hires.dat")
hires$=MidStr(ReadBytes(2, size_hir),2,8000,
#ENCODING_RAW)
WriteBytes(1, hires$, 8000)
CloseFile(2)

;open screen.dat and add it to .kla, after
removing the initial address
OpenFile(3, "screen.dat")
screen$=MidStr(ReadBytes(3, size_scr),2,1000,
#ENCODING_RAW)
WriteBytes(1, screen$, 1000)
CloseFile(3)

;open colors.dat and add it to .kla, after
removing the initial address
OpenFile(4, "colors.dat")
colors$=MidStr(ReadBytes(4, size_col),2,1000,
#ENCODING_RAW)
WriteBytes(1, colors$, 1000)
CloseFile(4)

;add last byte to .kla (background color)
- total 10003
WriteByte(1, ToNumber("00",16))
CloseFile(1)

```

A questo punto abbiamo realizzato tutti gli strumenti necessari al nostro scopo ed è quindi il momento di testarli. Nello specifico, ho scelto di lavorare su un'immagine presa da un vecchio anime degli anni 70: Babil Ni Sei (Babil 2°, conosciuto in Italia come Babil Junior).

Il procedimento è stato il seguente:

- ho scaricato un'immagine da internet, l'ho convertita in bitmap 320x200 e processata con il programma di Marco (vedi RMW #14). In Fig. 8 potete vedere l'immagine originale e, in Fig. 9, quella processata e caricata sul C64;

- usando i file .dat ho ottenuto, tramite il mio codice, un file .kla che ho caricato in Multipaint (Fig. 10);
- in seguito ho corretto e rielaborato l'immagine con Multipaint, processo non ancora concluso (Fig. 11). I dati immagine potranno essere in seguito estratti nuovamente per inserirli in un progetto completo per C64/128.

Siamo arrivati anche questa volta alla conclusione del nostro viaggio e, ancora una volta, abbiamo un nuovo strumento a disposizione per le nostre avventure nel "retro-verso". Come sempre, la soluzione presentata non ha la pretesa di essere la migliore ma spero abbia solleticato la vostra voglia di sperimentare e di tenere viva la vostra passione. Arrivederci sul prossimo numero!

RISORSE

<http://multipaint.kameli.net/>
<https://en.wikipedia.org/wiki/KoalaPad#KoalaPainter>
<https://www.hollywood-mal.com/>
https://github.com/marcus73/retromagazine_03
https://github.com/marcus73/retromagazine_04
<http://commodore128.mirkosoft.sk/vic-iie.html>
https://it.wikipedia.org/wiki/Babil_Junior

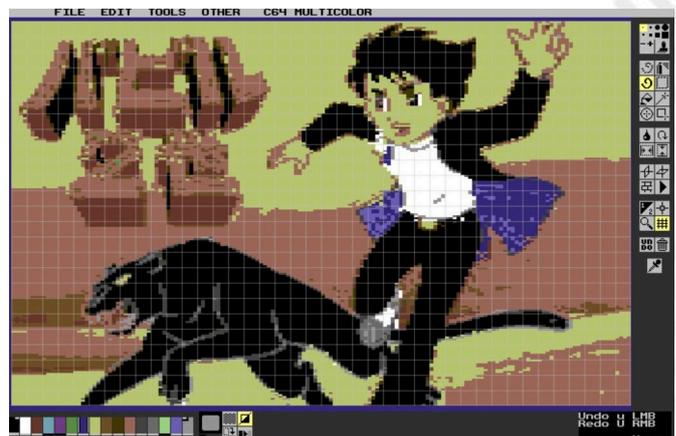


Fig. 10 - Babil junior originale



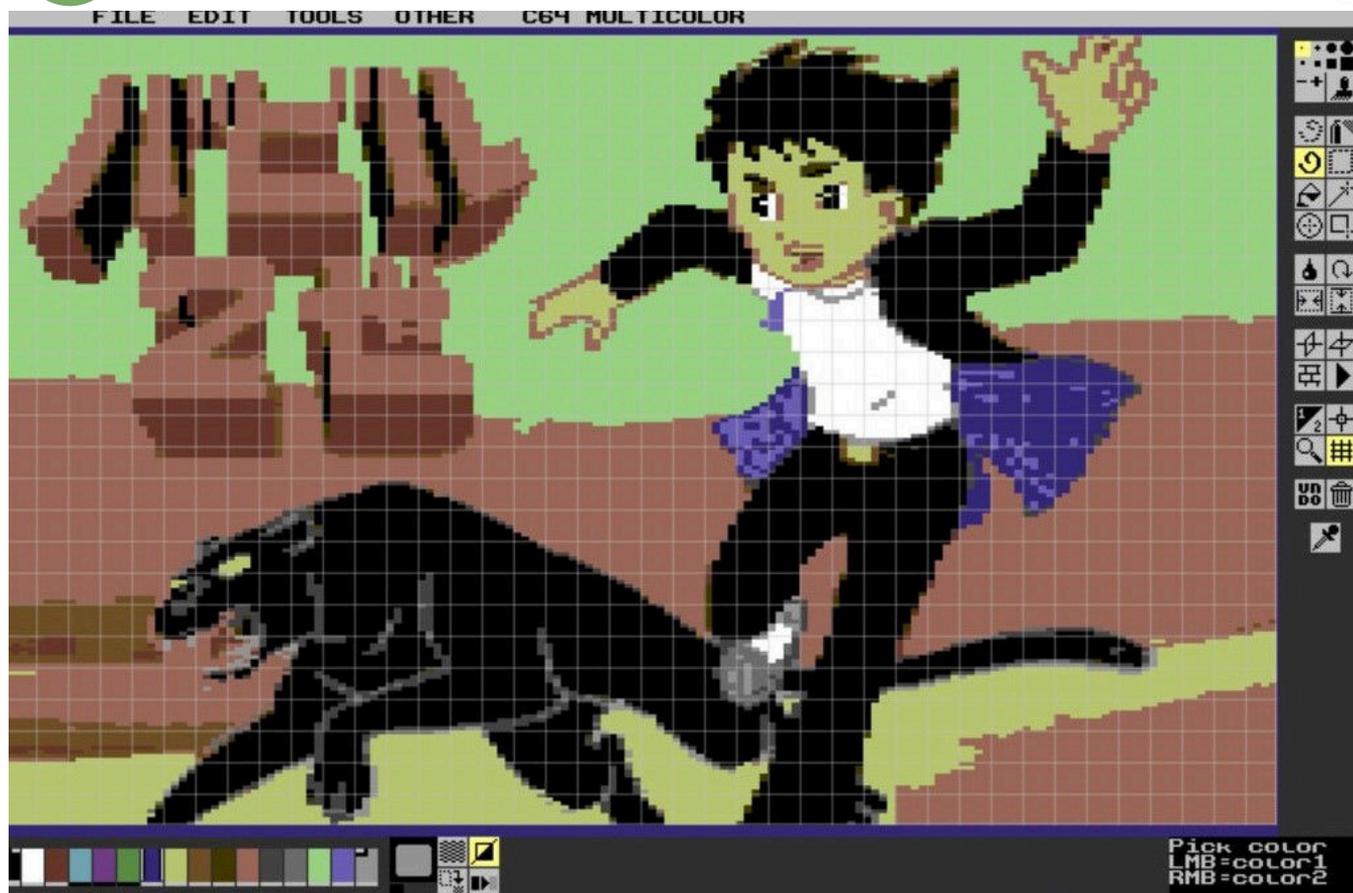


Fig. 11 - Babil junior ricolorato

BIBLIOGRAFIA

Grafica... Che passione! RMW#13 Marco Pistorio
Grafica ... Che Passione! Parte 2 RMW#14 Marco Pistorio
Introduzione alla grafica su Commodore C128
RMW#21 Gianluca Girelli
Grafica ... Che Passione! Parte 3 - RMW#40 di Gianluca Girelli, con il contributo di Marco Pistorio
C64 Programmer's Reference Guide - Copyright (C) 1982 by Commodore Business Machines, Inc.
Compute's Mapping the 64 and 64C - Sheldon Leemon, Compute! Publications Inc 1984,1987
C128-128D System Manual(IT) - Commodore Italiana SpA, 1985
C128 Service Manual - CBM, Inc 1987
C128 Oltre il Manuale - Rita Bonelli, Edizioni Jackson 1986
Mapping the C128 - Ottis R. Cowper, Compute! Publications Inc. 1986

NOTE

[1] Il processore è "little-endian" quindi il byte che viene memorizzato prima nelle locazioni designate come indirizzo è quello meno significativo (least significant) e il byte che viene dopo è quello più significativo (most significant). Anche se visivamente a noi esseri umani sembra che i byte siano in ordine inverso, in realtà ha senso che il byte meno significativo si trovi a un indirizzo minore e quello

più significativo a un indirizzo maggiore. Nei processori "big-endian" avviene il contrario. Fonte: Simone Bevilacqua, "Le basi della programmazione".

RINGRAZIAMENTI

L'autore desidera ringraziare in modo particolare:
Tero Heikkinen (Dr. TerrorZ), per Multipaint e per lo screenshot MSX;
Davide Bucci, per lo screenshot C64, preso da "The Queen's Footsteps";
Stefan Vogt, per lo screenshot ATARI preso da "The Curse of Rabenstein".





Ottimizzazione MSX BASIC e supporto Assembly

di Germán Gómez Herrera - Ingegnere di sistemi - Spagna

Nell'ultimo articolo dedicato a questo argomento, abbiamo parlato di come migliorare le prestazioni della velocità di esecuzione quando si scrive un programma in linguaggio Basic.

Questa volta cercheremo di migliorare le prestazioni di un programma Basic utilizzando l'Assembly, a volte erroneamente detto anche codice macchina, dove il programma Basic può richiamare qualsiasi routine in Assembly.

Riprendiamo l'esempio di gioco dell'ultimo articolo. Il gioco completo è giocabile, ma se ricordate, lo sfondo era uno stage statico.

La domanda è: non sarebbe meglio uno sfondo dinamico? E se così fosse, potremmo realizzare uno sfondo dinamico? Ad esempio, uno sfondo grafico a scorrimento. La risposta è sì, e il modo per farlo è riportato di seguito.

Un primo approccio, sarebbe quello di scrivere uno scroll rotazionale. Questo è facile nel sistema MSX (1) perché la routine deve solo gestire la tabella dei nomi (si veda il listato sotto) e inoltre occupa pochissimo spazio in memoria (2).

```
table equ 01800h + 32
buffer      equ 0c100h
v2r      equ 059h
r2v      equ 05ch
```

scroll:

```
ld hl,buffer
ld b,23
```

loop:

```
push bc
```

```
ld a,(hl)
ld d,h
```

```
ld e,l
inc hl
ld bc,31
ldir
```

```
ld (de),a
```

```
pop bc
```

```
djnz loop
```

```
ld hl,buffer
ld de,table
ld bc,768-32
```

```
call r2v
```

```
ret
```

setTable:

```
ld hl,table
ld de,buffer
ld bc,768-32
call v2r
```

```
ret
```

Ma questo metodo mostra un effetto visivo semplice e non realistico. Infatti, quando si osservano da lontano gli oggetti in movimento, si può notare che si muovono con velocità diverse. Questo si chiama effetto di parallasse (3).

Effetto parallasse

Sappiamo già cosa dobbiamo fare, ma come lo codifichiamo? Questa routine è più complessa della precedente.

Da un lato, la routine deve gestire ogni oggetto, in questo caso le stelle, una per una. Dall'altro, dobbiamo usare qualcosa come gli indici per puntare a ciascun oggetto.



**Fig. 1 - Fondale con 8 stelle**

Scrivere il programma in assembly può essere piuttosto difficile e quindi è più comodo usare un compilatore.

Se si utilizza un compilatore, le prestazioni della programmazione saranno molto più efficaci.

Inoltre, il compilatore può essere utilizzato in modo ottimale. Ad esempio, si possono usare tabelle invece di costrutti condizionali, non usare variabili di tipo struct ma vettori individuali e così via. Questi consigli miglioreranno la velocità di esecuzione della routine in codice macchina.

Il linguaggio compilato scelto è il C (4) perché è più vicino al computer. Ad esempio, è possibile utilizzare i puntatori per puntare all'indirizzo della memoria.

In questo linguaggio, abbiamo scritto tre funzioni: la prima per impostare gli oggetti, un'altra per spostarli e l'ultima è una funzione ausiliaria.

Inoltre, abbiamo esaminato il codice assembly in uscita per correggerlo.

Sulla base dell'ultimo consiglio, è molto meglio usare le tabelle invece dei costrutti condizionali; sotto vi mostro un esempio in cui ogni stella è dipinta con un colore basato sulla sua distanza in profondità.

Per spostare le stelle, si può assegnare uno spostamento in questo modo:

```
color=tcoll[i];
x[i]-
=(8*(color==15)+4*(color==5)+2*(color==4));
```

Dove colore è il codice colore della stella.

Ma è più opportuno utilizzare le tabelle in questo altro modo:

```
int tcol[3] = { 15, 5, 4 };
int tspeed[3] = { 8, 4, 2 };
x[i]-=velocità[i]);
```

In quest'ultimo modo, la dimensione del codice assemblato è inferiore e quindi le prestazioni della velocità di esecuzione saranno superiori.

Il codice completo è riportato nel listato qua sotto:

```
int nstar=16;

int tcol[3] = { 15, 5, 4 };
int tspeed[3] = { 8, 4, 2 };
int i, ctmp;
int *px, *py;

int x[32], y[32], col[32], speed[32];

initStars() {
    for(i=0; i<nstar; i++) {
        x[i]=rnd(256);
        newStar();
    }
}

moveStars() {
    for(i=0; i<nstar; i++) {
        pset(*(px=&x[i]), *(py=&y[i]), 1);
        if ((*px=*px-speed[i])<0) {
```

**Fig. 2 - Fondale con 16 stelle**



```

    *px=255;
    newStar();
}
pset(*px,*py,col[i]);
}
}

newStar(){
    y[i]=rnd(180)+10;
    col[i]=tcol[ctmp=rnd(3)];
    speed[i]=tspeed[ctmp];
}

```

Una volta ottenuto il codice in Assembly non resta che codificarlo nei dati in Basic e chiamare la routine dal programma Basic tramite il giusto indirizzo.

Inoltre, è necessario considerare il numero di stelle che la routine deve gestire. Il numero è sedici per impostazione predefinita, ma è possibile modificarlo fino a 32 stelle gestendo la variabile ST. Inoltre, si deve notare che se ci sono molte stelle, la velocità di esecuzione della routine sarà inferiore.

Le differenze sono visibili nelle figure da 1 a 3:

- Figura 1. Fondale con 8 stelle
- Figura 2. Fondale con 16 stelle
- Figura 3. Fondale con 32 stelle

Nel complesso, il risultato è il seguente: Un gioco completo con scorrimento in parallasse e fino a quattro sprite contemporaneamente in gioco come mostrato nella figura 4.



Fig. 3 - Fondale con 32 stelle



Fig. 4 - Gioco a scorrimento parallasse

L'immagine disco autoeseguibile con il gioco è allegata come di consueto:

https://retromagazine.net/download/msx_demo_2.dsk

Come spesso si dice:

La pratica rende perfetti.

Buon divertimento!

Riferimenti citati nell'articolo

1. <https://en.m.wikipedia.org/wiki/MSX>
2. <https://map.grauw.nl/articles/scrolling.php>
3. <https://en.m.wikipedia.org/wiki/Parallax>
4. [https://en.m.wikipedia.org/wiki/C_\(programming_language\)](https://en.m.wikipedia.org/wiki/C_(programming_language))





Due parole con il genio di Kobe

di Takahiro Yoshioka – trad. Carlo Nithaiah Del Mar Pirazzini

Il fondatore di Microsoft Bill Gates una volta disse: "Di tutte le persone che abbia mai incontrato, Kay (Kazuhiko) è probabilmente la più simile a me".

Kazuhiko Nishi, nato a Kobe, in Giappone, nel 1956, è uno dei pionieri dell'era che ha portato alla nascita dei personal computer.

Un tempo era a capo di Microsoft in Giappone ed è l'inventore dello standard MSX, uno dei primi e più amati computer degli anni '80.

A 66 anni, Nishi-San non ha intenzione di rinunciare al suo impegno per l'innovazione. È attivo in molti progetti, dalla costruzione di supercomputer più economici alla fondazione di un'università in Giappone.

L'Università Aperta della Catalogna ha recentemente invitato Nishi a Barcellona, nell'ambito delle celebrazioni per il 25° anniversario del programma di studi in Informatica, Multimedia e Telecomunicazioni.

Lì, ha presentato il modello aggiornato del microcomputer MSX, più di 30 anni dopo che è diventato disponibile al grande pubblico.

Oggi Nishi è una star, una fonte d'ispirazione per tutti gli appassionati di programmazione e ingegneria informatica. Imprenditore di successo che trova la sua vera vocazione nel giocare con le righe di codice, ha l'aura profetica di chi ha capito, prima di chiunque altro, che la tecnologia deve avvicinarsi il più possibile alle persone.

Nell'intervista abbiamo appreso da Mister Nishi alcune informazioni aggiuntive che non erano ancora state rivelate fino ad ora. Ci auguriamo che questo articolo serva da punto di riferimento per coloro che esitano ancora nel sostenere questo crowdfunding.

Taka: - Mr. Nishi per me è una specie di leggenda. Un creatore ed innovatore che per 30 anni ha lavorato nel settore PC/IT Perché ha smesso di sviluppare MSX 30 anni fa?

Nishi: - Nel 1983, IBM stava promuovendo il proprio computer, ma era ancora molto costoso e ho sentito il



Fig. 1 - MR. Nishi e il prototipo del MSX0

bisogno di creare una soluzione domestica più economica. Ecco come è nato MSX. Tuttavia, ero in Microsoft e un'azienda non può supportare due prodotti concorrenti. Pertanto, la macchina IBM è diventata popolare perché era più potente, quindi gli investimenti sono aumentati e i prezzi sono diminuiti. Nel 1993, MSX non era più redditizio, quindi ho smesso di svilupparlo.

T: - Perché il suo prodotto non è diventato lo standard internazionale a cui aspirava?

N: - Gli Stati Uniti erano il mercato dei computer Commodore. La Gran Bretagna era di [ZX] Spectrum. Sono stato molto fortunato in Spagna, Olanda, Italia e America Latina. Mi occupavo del design e i designer in genere non si preoccupano del marketing. Ora sono responsabile di tutto per il prossimo modello di MSX e viaggerò per tutta Europa per parlare del progetto.

T: - Il successo del retrogaming e il nuovo mercato dei nostalgici hanno qualcosa a che fare con il suo ritorno?

N: - No, ci sono cose nuove qui. Nel 1980 nasce il personal computer. Nel 2000, lo SMARTPHONE. Ogni 20 anni nasce un nuovo tipo di computer. Dal 2020 è il momento dell'IoT (Internet of Things). La mia idea è quella di produrre un MSX molto compatto che sia collegabile ai sensori IoT: questo è il nuovo MSX0, al prezzo di circa \$150. L'anno





prossimo rilascerà l'MSX3, che sarà collegabile al televisore. E poi, l'MSX Turbo, che sarà un super-computer. Voglio offrire una soluzione più semplice rispetto ai nostri concorrenti. Uso il linguaggio [di programmazione] BASIC, che è vecchio, ma ancora valido e facile da usare. Inoltre, non mi piacciono i videogiochi.

T: - Dopo tanti anni, la sua passione non sembra venir meno. Qual è il segreto?

N: - Tutti nascono con una missione data da Dio... ma nessuno sa cosa sia. Certo, sarebbe più facile nascere con un pezzo di carta che ti dice quale sarà la missione! Invece siamo tutti in viaggio per trovarlo. Non so se è la mia missione, ma forse il mio talento è l'ingegneria. In Microsoft ho fatto due grandi cose: MS-DOS (Microsoft Disk Operating System) – con estensione GW-BASIC e BASIC – e definire la tastiera e il mouse di Windows, oltre a collegare il CD-ROM. Questo è tutto merito mio.

Dopo Microsoft, ho fatto molti soldi con la mia azienda lavorando sulle CPU e ho anche fondato la comunità mondiale per il formato di compressione video MPEG. Quando ho compiuto 60 anni, mi sono chiesto cosa avrei fatto dopo. Ho capito che la mia prossima sfida sarebbe stata l'IoT.

T: - Si è mai pentito di aver lasciato Microsoft nel 1985?

N:- Nella vita ci sono cambiamenti positivi o negativi, ma

sono sempre una tua decisione. Ero molto vicino a Bill Gates: controllavo il Giappone per conto dell'azienda e facevo solo rapporto a lui. Ma c'erano altri burocrati che volevano prendere il mio posto. Spesso, quando le aziende diventano grandi, si riempiono di burocrati ambiziosi. Ho deciso di non combattere e me ne sono andato. Se fossi rimasto in Microsoft, ora avrei almeno un decimo di ciò che possiede Bill Gates. Ho sofferto per circa 10 anni dopo quella decisione. Ma poi mi sono ripreso e ho reso pubblica la mia azienda. Ho guadagnato circa 300 milioni di dollari: non è abbastanza? Per un po' ho pensato che non fosse abbastanza, ma lo era. Voglio dire, ho quattro elicotteri, una Rolls-Royce, una Bentley. Quindi, sono felice.

T: - È questa la sua idea di felicità?

N: - Tutti sono alla ricerca della felicità. E poi le persone sentono che i soldi possono comprare la felicità. Ci sono tanti modi di essere felici che ho scoperto: realizzare il proprio sogno, essere famosi, fare un buon lavoro, diventare famosi. Tonnellate di denaro. Ma se cerchi solo soldi, potere e fama, non hai limiti. Con i beni, devi essere mentalmente consapevole, devi avere gratitudine. I monaci in Tibet sono più felici delle persone che vivono con molto. Forse la coesistenza di queste cose può rendere le persone estremamente felici.

T: - Tornando al presente: cosa pensa della crisi delle



Fig. 2 - Nishi, Bill Gates e Paul Allen





Fig. 3 - MR. Nishi

Big Tech?

N: - Elon Musk ha licenziato tre quarti dei dipendenti di Twitter, ma l'azienda è ancora in funzione. Tutti sanno che Big Tech è piena di lavoratori pigri, perché le aziende sono così redditizie che se lo possono permettere. Il management è stato molto generoso nel corso degli anni: che tu lavori o meno, non importa, rilassati. Ora, a causa dell'estrema disciplina e dell'atteggiamento di Elon Musk, queste aziende stanno licenziando le persone come un gesto, per dimostrare che stanno facendo uno sforzo. La verità è che in ogni azienda, scuola o comunità, almeno il 10% del personale è improduttivo.

T: - Cosa ne pensi di ChatGPT?

N: - Il mio professore al MIT, Marvin Minsky (uno dei pionieri nella ricerca sull'IA), ha sempre detto che è meglio non fare affari basandosi su AI. Ma ora sta diventando molto popolare. Il paese più avanzato è la Cina, mentre gli Stati Uniti cercano disperatamente di recuperare. Non so cosa accadrà, ma ChatGPT è ancora molto, molto stupido. Quando leggo i testi che produce, rido sempre. Detto questo, è solo questione di tempo... le macchine diventeranno molto intelligenti tra 10 o 20 anni. Personalmente, però, non sono molto interessato. Mi restano ancora circa 20 anni di lavoro e voglio dedicarne metà all'IoT e metà ai super-computer.



Fig. 4 - L'MSX0

T: - Cosa ha in mente per i super-computer?

N: - Molte persone leggono e sentono parlare di super-computer, ma quasi nessuno li ha toccati, perché sono incredibilmente costosi. Se potessi produrne uno al prezzo di un'auto, tutti potrebbero usarlo. Forse non ne avremmo uno in ogni casa, ma ne avremmo uno in ogni università o laboratorio. Per inciso, i super-computer usano ancora il linguaggio Fortran 77, che ha 45 anni. Dobbiamo avvicinarli alle persone, in modo che possano pensare a nuovi linguaggi per l'IA.

T: - Sta ancora imparando?

N: - Ho dovuto abbandonare l'università perché dovevo avviare la mia azienda. Quando ho potuto ho provato a rientrare, ma il sistema universitario non lo ha permesso. Alla fine, ho terminato il mio dottorato di ricerca all'età di 60 anni e mi è stato offerto un posto di insegnante presso l'Università di Tokyo. Insegno da cinque anni. Ora devo andare in pensione, ma ho deciso di costruire la mia università investendo tutti i soldi che ho. Spero di raggiungere questo obiettivo entro il 2025: sarà la mia più grande sfida. L'innovazione non dipende dall'età fisica, ma dall'età mentale e dal cuore. La giovinezza non è un periodo specifico della vita: puoi rimanere giovane cambiando il tuo modo di pensare.





ListAmiga - I 10 MIGLIORI PLATFORM SU AMIGA

di Beppe Rinella

Carissimi fratelli e carissime sorelle che, come me, bevete bibite a trovate in giro nella speranza che vi doni incredibili poteri, bentornati alla consueta ListAmiga!

La nostra cara e vecchia Amiga ci ha donato centinaia di giochi meravigliosi, di generi molto diversi tra loro. Alcune tipologie di giochi trovarono particolare fortuna sulla macchina di mamma Commodore, tra queste ci sono senza dubbio i platform game.

Amiga ha offerto a tutti noi un gran numero di platform di qualità davvero eccelsa, tanto che stilare questa lista è stato per me davvero difficile, gli esclusi illustri sono davvero tanti, forse troppi!

Il concetto del platform è davvero molto semplice e basilare e può essere riassunto più o meno così: impersoniamo un personaggio che deve arrivare alla fine del livello, saltando su piattaforme, evitando nemici e pericoli di varia natura.

La bellezza di tanta semplicità è la possibilità di interpretare il genere in modi molto diversi tra loro. Può essere sviluppato puntando più sull'avventura, sull'esplorazione o lo shooting. Il mood può spaziare dall'allegria totale, con livelli pieni di colori e personaggi buffi, a un'atmosfera più seria.

Nella lista che vi propongo ho cercato di includere non solo i titoli migliori in quanto a qualità, ma cercando di includere titoli che mostrino le diverse declinazioni possibili per questo genere.

Partiamo quindi con

I 10 MIGLIORI PLATFORM SU AMIGA

FIRE AND ICE

Iniziamo con il botto.

In Fire and Ice saremo chiamati ad impersonare Cool Coyote, un Coyote blu che dovrà affrontare 7 livelli con ambientazioni molto diverse tra loro. Per portare a termine ogni livello sarà necessario trovare i diversi pezzi della chiave necessaria per poter aprire l'uscita. Ogni pezzo è custodito da uno dei nemici presenti nel livello, ma non sapremo quali, motivo per cui ci toccherà eliminarli tutti. Per farlo la nostra unica arma sono delle palline di ghiaccio con cui potremo momentaneamente ibernare il malcapitato



Fire and Ice

di turno, per poi poterlo comodamente frantumare in mille pezzi!

Inizieremo la nostra avventura nel livello "artico", per poi passare alla campagna, l'interno di un castello, il livello subacqueo, la foresta pluviale, un tempio azteco e infine l'antico Egitto dove dovremo affrontare il cattivone Suten. I livelli sono decisamente estesi e tutti da esplorare, pieni zeppi di nemici e pericoli assortiti.

Graficamente Fire and Ice è una vera delizia, a partire da Cool Coyote animato perfettamente, passando poi per i nemici carini e simpatici solo nell'aspetto. Le ambientazioni poi sono perfettamente caratterizzate e varie.

Nella parte inferiore dello schermo poi è visibile una piccola mappa che ci mostra a che punto della nostra avventura ci troviamo, con un effetto riflesso sull'acqua che è un vero tocco di classe.

Fire and Ice è una gioia da giocare, preparatevi però a un livello di difficoltà non esattamente benevolo, niente di eccessivo ma piuttosto impegnativo.

Ho sempre voluto un gran bene a Fire and Ice e gliene voglio ancora, è un gioco davvero squisito e sprizza amore da tutti i pori!

SOCCER KID

Ne abbiamo parlato su queste pagine nello scorso numero, nella lista dei migliori giochi di calcio, nonostante sia un platform, e che platform.

Nei panni del nostro piccolo campione dovremo recuperare i pezzi della coppa del mondo (finita in frantumi durante il tentativo di furto da parte di alieni collezionisti di trofei) sparsi in cinque diversi paesi: Inghilterra, Italia, Russia,





Soccer KID

Giappone e USA.

Per portare a termine la nostra missione avremo come arma il nostro fidato pallone con cui percuotere i nemici che tenteranno di ostacolarci. Il protagonista è un vero fenomeno con il pallone tra i piedi, potremo quindi eseguire palleggi, rovesciate, colpi di testa, e altro ancora oltre a poter utilizzare il nostro pallone come trampolino per eseguire dei gran balzi.

Ogni paese che ci troveremo a visitare è composto da tre livelli che si concludono con l'immane boss finale. Sparse per i diversi livelli potremo inoltre raccogliere le figurine, undici per ogni paese, completando la collezione avremo accesso a livelli bonus.

Soccer Kid ha davvero tutto, graficamente non perde un colpo, il nostro eroe è ricco di animazioni, ambientazioni e nemici sono ottimamente realizzati e caratterizzati, il tutto scorre senza la minima incertezza. Lungo e impegnativo quanto basta e dannatamente divertente.

Insomma, non c'è davvero alcun motivo per non giocare e rigiocare Soccer Kid!

SUPERFROG

Non si può parlare di platform su Amiga senza citare Superfrog e infatti eccolo qua.

La storia racconta della più classica delle streghe che trasforma in rana il più classico dei principi, rapendo anche la altrettanto classica principessa. Il nostro (non ancora) eroe, triste e sconsolato, ingurgita una bibita (Lucozade, sponsor del gioco) grazie alla quale diventa Superfrog, con tanto di mantello rosso svolazzante e via a salvare la nostra amata.

Niente di particolarmente originale ma non serve esserlo quando, impugnando il joystick, ci si trova davanti un titolo di tale pregevole fattura.

Considerato (e presentato, almeno dalle riviste di settore di allora) la risposta a Mario e Sonic su Amiga, non è né uno né l'altro, è Superfrog ed è una meraviglia così.



Superfrog

Bello da vedere, grazie a una grafica che è una delizia e senza sbavature, le musiche di un Allister Brimble davvero in forma sono un piacere da ascoltare.

La giocabilità è sopraffina e il livello di sfida fortunatamente più basso dello standard Team17. I livelli che ci troveremo ad affrontare (6 mondi per un totale di 24 livelli) sono estesi e ricchi di segreti da scovare, il livello bonus con la slot machine è un intermezzo tra un livello e l'altro davvero gradevole, il livello sparattutto "Project F" un vero tocco di classe.

Superfrog è un inno alla semplicità e purezza del genere platform, non vuole essere niente di più, non cerca di rivoluzionare niente. Tutto ciò che fa però lo fa dannatamente bene.

Impossibile giocare a Superfrog senza un bel sorriso stampato in volto.

TURRICAN 2

Un capolavoro assoluto, sono assolutamente certo che se stai leggendo queste righe tu l'abbia giocato e tanto anche. La trama, come spesso succede, è abbastanza semplice



Turrican 2





e classica: l'imperatore galattico "The Machine" distrugge l'astronave Avalon 1 su cui si trova il nostro eroe Bren McGuire, unico superstite sopravvissuto che, indossando l'armatura Turrican parte in cerca di vendetta.

Turrican 2 è un platform decisamente diverso da quelli visti fino ad ora (e che vedremo in seguito), qui si spara e non si fanno prigionieri, grazie anche al potenziale offensivo di cui siamo dotati. Lungo il nostro cammino troveremo e potenziemo armi devastanti, tenendo premuto il bottone del nostro fidato joystick partirà una serie di colpi in rapida successione che formano un potentissimo raggio che potremo direzionare a 360 gradi, senza però poterci muovere. Potremo poi trasformarci in una devastante palla rotante indistruttibile che semina mine altamente distruttive. Unica pecca di questa configurazione è la scarsa maneggevolezza (voluta ovviamente, altrimenti sarebbe davvero troppo semplice!). La possibilità di sparare un laser che stermina tutti i nemici presenti a schermo, oltre alla modalità "berserk", cioè configurazione "palla" e devastazione totale grazie all'uso simultaneo di qualunque arma possibile!

Nella materia "level design", Turrican 2 sarebbe tranquillamente argomento di esame. I 12 livelli, suddivisi in 5 ambientazioni differenti sono enormi, non solo azione dunque ma tanta esplorazione che richiede sì una certa dedizione, ma regala soddisfazioni come pochi giochi sanno fare, parlo al presente non a caso.

Graficamente, Turrican 2 rasenta la perfezione e non mostra la minima incertezza, anche in fasi in cui il numero di nemici a schermo è davvero notevole. La varietà è poi garantita anche grazie ai livelli shot'em up di ottima fattura, con un piccolo omaggio al classico R-Type mentre partiamo con la nostra astronave. Sono cose che scaldano il cuore.

Il sonoro merita un capitolo a parte, il nome Chris Huelsbeck è garanzia di qualità eccelsa, le musiche che ci accompagnano nel corso della nostra avventura sono semplicemente meravigliose, ascoltare la OST per credere. Turrican 2 riesce nel non facile compito di migliorare il suo predecessore sotto ogni aspetto, impresa non facile data l'altissima qualità del primo capitolo.

Ci troviamo al cospetto di un autentico capolavoro, l'ho già detto ma lo ripeto, un pezzo di storia del videogioco, non solo su Amiga.

JAMES POND 2 - CODENAME ROBOCOD

Getto la maschera e lo dico subito, questo è uno dei miei videogiochi del cuore, tanti sono i magnifici ricordi che in me fa riemergere. Ma non è dovuta a ciò la sua presenza in questa lista, perchè James Pond 2 è davvero un gran bel platform.



James Pond 2

Partendo dalla storia, anche in questo caso nulla di eccezionale: il Dr. Maybe, villain del primo capitolo, prende il controllo della fabbrica di giocattoli di Babbo Natale, imprigionando i poveri pinguini. Il nostro compito sarà quello di liberarli girando in lungo e in largo per i 50 livelli di cui il titolo si compone.

James Pond 2 è un platform semplice ed essenziale, tutto ciò che potremo fare per contrastare i tanti nemici presenti, sarà evitarli o saltarci sopra in pieno stile Mario. L'armatura che abbiamo in dotazione (da qui il riferimento a "Robocop" del titolo, la cui colonna sonora riarrangiata ritroviamo nella presentazione) ci permette inoltre di allungarci a dismisura verso l'alto, fino a quando troveremo una piattaforma a cui aggrapparci, permettendoci così di raggiungere vette indicibili.

Nonostante il gameplay possa sembrare fin troppo banale (trova il pinguino - esci dal livello), la varietà è garantita dalle dimensioni dei livelli, grandi e pieni di segreti da scoprire, insieme a power ups di vario genere come le ali per...beh, volare, l'ombrello per planare oltre a una serie di veicoli che potremo utilizzare qua e là. Macchine, aeroplani e una formidabile vasca volante!

Graficamente James Pond 2 è una vera delizia, pur non essendo altamente spettacolare è tutto adorabile, compresi i nemici che in realtà adorabili non sono. Lo stesso si può dire delle musiche, ottime e perfettamente in tema.

I livelli sono vari e comprendono diverse ambientazioni: i dolciumi, gli orsacchiotti, la musica e tanto altro ancora. I controlli sono semplici e precisi, il livello di difficoltà non è troppo alto ma portare a termine l'avventura del pesce-robot è tutt'altro che semplice, considerata anche la lunghezza e la mancanza di salvataggi o password che ci consentano di non dover ricominciare tutto da capo ogni volta che inseriamo il floppy (tanta bellezza condensata in poco più di 1Mb!).

James Pond è pura gioia, semplice, immediato, impegnativo, colorato e divertentissimo, cosa si può chiedere di più da



**Leander**

un platform degno di questo nome?

LEANDER

Ed eccoci arrivati alla clamorosa opera prima di Traveller's Tales, Leander, un prodotto di una qualità sopraffina.

Ormai l'abbiamo capito, quando parliamo di platform la trama non è certo ciò che conta di più, molto brevemente: nei panni del cavaliere/principe Leander dovremo affrontare e sconfiggere il malvagio Thanatos per liberare la nostra amata principessa.

Leander unisce alle piattaforme proprie del genere, una massiccia dose di avventura ed esplorazione. Ci troveremo così a girare in lungo e in largo i tre mondi suddivisi in 21 livelli totali, facendoci largo tra nemici agguerriti e altamente spettacolari, alla ricerca di un oggetto specifico che ci permetterà di passare al livello successivo.

Saremmo così immersi nelle meravigliose ambientazioni fantasy di questo capolavoro, ricche di atmosfera e altrettanti pericoli, accompagnati della nostra fidata spada con cui affronteremo ogni sorta di creatura fantastica. In giro per i livelli troveremo preziose monete spendibili nei negozi che incontreremo, dove potremo acquistare armature ed armi sempre più potenti.

Tecnicamente Leander è un gioiello sotto ogni punto di vista, graficamente non ha la minima sbavatura, ci troveremo spesso a fronteggiare mostri di dimensioni colossali quanto letali, le ambientazioni fortemente evocative fanno il resto, creando un'atmosfera davvero immersiva.

Sul fronte audio, l'unico difetto è che saremo obbligati a scegliere tra musiche o effetti sonori, questi ultimi fanno egregiamente il loro dovere, mentre le musiche sono opera del magnifico trio Matthew Simmonds, Timothy Wright e Jon Burton, autori delle grandiose sinfonie che accompagnano il nostro eroe.

Vivere l'avventura di Leander è un vero piacere, la difficoltà è ottimamente calibrata, i ritmi mai frenetici quanto piuttosto ragionati ma non per questo privi di scontri epici. Leander è un titolo da vivere almeno una volta nella vita,

ma una volta terminato scommetto che non resisterete alla voglia di tornare dentro il suo meraviglioso mondo.

RODLAND

Esiste un genere di platform il cui indiscusso capostipite è Bubble Bobble, le cui caratteristiche fondamentali sono i livelli a schermata singola dove eliminare i nemici che lo popolano, così da passare al livello successivo, ripulire ancora e così via.

Rodland è praticamente un Bubble Bobble più recente rispetto allo storico padre spirituale, delizioso e piuttosto impegnativo.

Anche in questo caso, sbrighiamo subito la questione trama, anche qui nulla di eclatante: Tam e Rit, le protagoniste, sono due fatine che vivono serene nel loro bellissimo villaggio incantato, fino a quando la loro povera madre viene rapita dai cattivoni guidati da un minotauro gigante, ovviamente le nostre eroine partiranno subito per salvare la madre imprigionata in cima a una torre.

Rodland è la conversione del Coin-Op apparso nel 1990 in sala giochi per poi essere convertito dal 1991 su diverse piattaforme. Su Amiga la conversione è davvero impeccabile, evento non certo frequente per le conversioni da Coin-Op per la macchina Commodore. Rispetto al Coin-Op il numero di livelli è inferiore, qualche piccolo rimaneggiamento grafico c'è ma nulla di eclatante.

Ci troveremo dunque ad impersonare la fatina Tam, con i suoi capelloni rosa e armata di bacchetta magica con cui eliminare i nemici che infestano i livelli. Per farlo potremo colpirli spingendoli indietro oppure imprigionarli in una piccola gabbia magica, per poi sbatterli a terra a destra e sinistra. Avete come Hulk massacra Loki nel primo Avengers? Ecco, così. Oltre ciò, le nostre fatine possono creare una scala da utilizzare come preferiamo, che si rivela in molti momenti un'ottima via di fuga.

Tecnicamente Rodland sprizza Giappone da tutti i pori, grafica coloratissima, tutto è molto carino e zuccheroso,

**Rodland**



BC Kid

anche le musiche sono più che appropriate per mantenere l'atmosfera da anime giapponese.

Giocabilità ottima, molto semplice ma per nulla facile, il livello di difficoltà infatti aumenta progressivamente fino a diventare piuttosto arduo, saranno molte le situazioni in cui tutto vi apparirà molto meno coccoloso, fidatevi. Da menzionare assolutamente la possibilità di giocare in doppio, il divertimento diventa epocale così come le imprecazioni, infatti con la nostra bacchetta potremmo accanirci contro la fatina (teoricamente) alleata sbattendola qua e là.

Rodland è un coloratissimo dolcetto, un bon bon irresistibile che garantisce abbondante divertimento misto a rabbia estrema!

BC KID

Altra conversione, questa volta non da Coin-Op ma da console, PC Kid (con la "P") è stata infatti la mascotte del glorioso PC Engine (macchina meravigliosa).

La conversione è opera dei ragazzi di Factor 5, autori dei tre episodi di Turrican su Amiga, dunque garanzia di qualità assoluta. Se poi aggiungiamo le musiche riarrangiate da Chris Huelsbeck, è facile capire il perchè questa conversione sia stata da molti considerata addirittura migliore dell'originale sotto molti aspetti.

In BC Kid impersoneremo Bonk, un piccolo cavernicolo dotato però di una testa enorme che sarà anche la nostra unica arma di offesa. Il nostro compito è quello di eliminare i dinosauri cattivi che popolano i diversi livelli fino ad arrivare al boss finale.

Come menzionato, tutto ciò che avremo a disposizione per sopravvivere sarà il testone del protagonista, e non mi riferisco alla materia grigia. Nel corso della nostra avventura infatti dovremo prendere a testate qualunque cosa si muova e vuole farci del male, cosa che personalmente mi ha dato sempre una certa soddisfazione.

Anche qui, un po' come Rodland, è tutto meravigliosamente cartone animato giapponese. Tutti i personaggi che popolano il mondo di BC Kid sono esteticamente stupidi e adorabili, il tutto mi ha sempre ricordato un po' Doraemon come estetica.

Graficamente pulito, colorato e senza la minima incertezza, le musiche assolutamente azzeccate poi sono un piacere per le orecchie.

La giocabilità è ottima, tutto è molto semplice come spesso un platform deve essere ma è divertente come poche cose. Bc Kid è uno di quei giochi che definisco "con il sorriso", impossibile non giocarlo sorridendo di continuo, almeno per chi come il sottoscritto è cresciuto a pane e anime giapponesi!

Un titolo assolutamente imperdibile per gli appassionati di piattaforme e non solo.

MR NUTS: HOPPIN' MAD

La storia di Mr Nutz su Amiga è nota: dato il declino di Amiga ormai inesorabile nel 1993, il titolo inizialmente previsto per la nostra macchina del cuore fu pubblicato per Snes e Megadrive. Si tratta di un buon platform, niente di eclatante ma più che dignitoso. Si decise però di pubblicare Mr Nutz anche su Amiga ma non si trattò di una banale conversione, bensì di un titolo completamente nuovo, escluso il protagonista tutto fu rifatto da zero e il gioco risultò essere completamente diverso dalle controparti su console. Non solo diverso ma migliore.

Nei panni di Mr Nutz, uno scoiattolo antropomorfo, saremo chiamati a salvare il mondo dalle galline aliene (don't ask). Questo gran bel titolo è la dimostrazione che per fare un grande gioco non serve sempre essere originali, anzi, in questo caso i ragazzi di Kaiko (Apidya dice niente?) attingono a piene mani dagli assoluti riferimenti dell'epoca in materia di platform: Mario World e (maggiormente) Sonic. Osservando la mappa tramite cui accedere ai diversi livelli, non è possibile non pensare a Mario World, con la differenza



MR NUTZ HOPPIN' MAD





che nel titolo Amiga la mappa è esplorabile alla ricerca di oggetti e personaggi con cui dialogare. Iniziato il primo livello invece, quindi passando al platform vero e proprio, la mente va subito a Sonic. Sia esteticamente che a livello di gameplay è evidente quanto il porcospino blu sia stato qualcosa in più di un'ispirazione. Il ritmo è molto alto (non supersonico, ma comunque veloce) e anche nei livelli bonus che rasentano la psichedelia è evidente l'ispirazione. Mr Nutz è tecnicamente sontuoso, con una grafica satura di colori estremamente vividi, uno scrolling perfetto, il tutto accompagnato da una musica ed effetti di ottimo livello. Il level design è notevole, i livelli enormi e tutti da esplorare, portare a termine Mr Nutz non è affatto semplice e veloce, richiede dedizione e tempo ottimamente speso.

Un titolo imperdibile che dimostra che a volte copiare non è poi così un male.

RUFF 'N' TUMBLE

Titolo che può essere considerato il canto del cigno della nostra amata Amiga, e che canto.

Uscito nel 1994, quando ormai il destino di Commodore era ampiamente segnato e le console a 32 bit facevano la loro comparsa, Ruff 'n' Tumble è un titolo assolutamente eccezionale.

Nei panni di Ruff Rogers, un bambino finito su un pianeta alieno, dovremo sconfiggere il potente tiranno di turno, oltre a recuperare le biglie smarrite.

Ruff 'n' Tumble fa del ritmo adrenalinico il suo punto di forza, grazie ad alcune ottime trovate di gameplay. Il protagonista dovrà infatti sparare a chiunque gli si pari davanti, raccogliendo le suddette biglie che ci permetteranno l'uscita dal livello. Le biglie non saranno le sole cose che dovremo raccogliere, troveremo infatti armi e power up che renderanno le nostre bocche di fuoco altamente devastanti, è proprio nel funzionamento delle armi che troviamo una gran bell'idea: le armi hanno munizioni infinite ma un limite di durata temporale, il che ci porterà a devastare il devastabile nel minor tempo possibile per sfruttare al massimo l'arma in dotazione.

Ruff 'n' Tumble è visivamente spettacolare, la grafica è di altissimo livello e tutto scorre liscio senza il minimo rallentamento, nonostante molto spesso lo schermo sarà pieno zeppo di nemici ed esplosioni. Le musiche sono altrettanto ottime, incalzanti e adrenaliniche quanto serve, gli effetti sonori sono perfetti.

Il level design poi è da manuale, i livelli sono enormi e ricchi di segreti, nonostante il gioco imponga una certa frenesia, ampio spazio viene comunque dato all'esplorazione e ricerca di segreti.

I quattro ambienti divisi in 16 livelli totali sono impegnativi, anche per via di una difficoltà non bassa ma mai ingiusta,



Ruff'n'Tumble

ogni 4 livelli affronteremo gli immancabili boss, enormi e spettacolari.

Ruff 'n' Tumble è titolo da giocare assolutamente, non concede un attimo di respiro e la vostra voglia di distruzione sarà ampiamente appagata.

Uscito fuori tempo massimo, Ruff 'n' Tumble è un atto d'amore verso la nostra cara Amiga e dimostra di cosa era davvero capace la macchina di mamma Commodore.

Cari amici e care amiche eccoci giunti al termine di questa ricca lista che, data la quantità e il livello dei tanti platform usciti su Amiga, potrebbe essere decisamente più lunga, sono tanti infatti i titoli che meritano di essere ricordati. L'invito dunque è sempre lo stesso, venite a trovarci sul nostro sito e sui nostri canali social e diteci la vostra lista.

Un saluto a voi tutti e AMIGA FOREVER!





NEW GAME

NEONNOIR

Anno: 2022/2023

Editore: Steamknight

Genere: Avventura grafica

Piattaforma: Amiga

Sito web: [https://](https://steamknight.itch.io/neonnoir)steamknight.itch.io/neonnoir

NEONNoir è un gioco d'avventura con tematiche cyberpunk realizzato da Steamknight apparso la prima volta lo scorso anno durante l'Amiga Blitz Basic Game Jam.

Il gioco suscita pareri contrastanti per via di una grafica impressionante, una bella atmosfera ma un gameplay che sembrava non completo. Ora riceve un nuovo aggiornamento che sistema alcuni punti "oscuri" e rende la storia più profonda.

Il titolo ci mette nei panni di un investigatore privato alla ricerca di una giovane donna con problemi di tossicodipendenza.

L'atmosfera che si respira è quella di Blade Runner o di Shadowrun (il gioco di ruolo).

Cosa hanno migliorato rispetto alla versione precedente? Sicuramente il motore di gioco risulta più fluido. Sono state sistemate le finestre di dialogo che apparivano confusionarie, hanno sistemato gli errori ortografici e alcune frasi non complete, la storia è più comprensibile e il tempo di gioco scorre in modo più fluido.

Il titolo è in sostanza un'avventura grafica punta e clicca con un sistema di inventario minimo.

È interamente giocabile con il mouse



e non è necessario cliccare su ogni oggetto e su ogni cosa per avanzare negli enigmi.

Il gioco richiede un A1200 con 2 Mb di chip ram, 4 o più mega di Fast Ram, un disco rigido con almeno 5 mb di spazio.

Funziona ovviamente in emulazione tramite i più noti emulatori. Sul sito itch.io degli sviluppatori è stato rilasciato anche il manuale per l'installazione corretta.

Il titolo è stato sviluppato interamente in Blitz Basic e C++ e il codice sorgente è disponibile sulla stessa pagina degli sviluppatori.

Tecnicamente è sicuramente ben fatto e con una bella musica d'atmosfera. Non è un titolo lungo e nemmeno complicato. In circa due ore di gioco risolverete il caso.

Per chi rimane incastrato nel gioco comunque è stata inserita un walkthrough.

Sicuramente un prodotto interessante per gli amanti del cyberpunk e dei giochi noir.

di Carlo Nithaiah Del Mar Pirazzini



GIUDIZIO FINALE

» Giocabilità 70%

Sistema punta e clicca e narrazione a schermo. Semplice e efficace.

» Longevità 60%

Non difficile e difficilmente lo rigiocherete.





NEW GAME

LESTER

Anno: 2023

Editore/Sviluppatore:

KNIFEGRINDER

Genere: Platform/Action

Piattaforma: Commodore 64

Sito web: [https://](https://knifegrinder.itch.io/lester)

knifegrinder.itch.io/lester



GIUDIZIO FINALE



» Giocabilità 95%

Comandi reattivi, level design ben fatto. Un gran bel gioco

» Longevità 90%

Sono 60 stanze e permette di giocare a lungo. Non difficile né impossibile ma divertente.

È proprio una brutta sensazione quando ti trovi da solo, armato con solo la tua piccola pistola laser e un'intera stazione spaziale fuori controllo.

Questo è capitato al povero androide L.3573-R, primo di una nuova generazione di guardiani indipendenti dalla AI della stazione spaziale.

Primo e più sfortunato in quanto l'intelligenza artificiale della nave è impazzita e ha deciso di portare il carico nucleare direttamente contro il pianeta terra per distruggerlo.

Quindi il compito di L.3573-R è proprio di raggiungere il centro nevralgico della nave, distruggere l'AI malvagia e salvare il mondo.

Un compito semplice sulla carta ma complicato in quanto la nave brulica di droidi nemici agguerriti e trappole mortali.

Il titolo si ispira ai moderni metroidvania usciti su MSX come Ghost (del 2017) e alle vecchie produzioni di Konami come The Maze of Galious.

Per raggiungere il centro di controllo della nave dovremo recuperare diversi chiavi d'accesso e districarci tra piattaforme e avversari.

Durante l'avventura è possibile potenziare il nostro armamento e recuperare i punti ferita persi attraverso dei terminali. Per poter acquistare energia e armi dovremo raccogliere le monete che ci rilasceranno gli avversari e gli oggetti abbattuti. Il terminale funge anche da mappa della nave e ci permette di capire dove ci troviamo dentro l'astronave.

All'inizio appare come un gioco semplice, ma mano a mano che ci addentriamo nei livelli la difficoltà cresce e sarà sempre più complicato andare avanti senza un corretto

potenziamento del proprio arsenale. Una piccola meccanica da rpg è stata inserita. Ogni volta che si distruggerà un robot avversario si alzerà la barra dell'esperienza che, una volta al massimo, ci permetterà di avere una salute migliore.

Lester è un gioco ben fatto. Solido, con controlli precisi e molto reattivi (perfetto se giocato con un joystick) che funzionano perfettamente con il gameplay ad un tasto solo.

La grafica è dettagliata, pulita e ben animata e ci ricorda quella di alcuni titoli per il Nintendo NES con una particolare cura nei dettagli (i fondali sono bellissimi).

Il comparto sonoro è bellissimo con il SID che ci sforna una colonna sonora bellissima e tracce sonore dal ritmo veloce perfette per mantenere l'azione al massimo.

Bella anche la piccola sequenza introduttiva. Lavoro magistrale.

Il titolo è compatibile con gli standard pal e ntc5 e dai nostri test funziona correttamente sia su real hardware, sia in emulazione che su The C64 family (mini e maxi).

Anche se ci sono 60 stanze da esplorare non è un titolo difficilissimo da portare a termine. Richiede impegno e un po' di manualità per i livelli più ostici, ma è godibile e per nulla frustrante.

Il titolo è disponibile gratuitamente nella pagina dello sviluppatore su itch.io. Vi consiglio però di lasciare una piccola mancia per questo lavoro perché lo merita.

Davvero un buon inizio per il Commodore 64 in questo 2023.

di **Carlo Nithaiah Del Mar Pirazzini**





NEW GAME

LIFE OF MARS

Anno: 2022

Editore: Kai Magazine

Genere: Action/Platform

Piattaforma: Sega Megadrive

Il gruppo dietro a Life of Mars è quello degli spagnoli Kai Magazine, già autori del divertente Metal Dragon e dello stesso Life of Mars per MSX.

Il responsabile del progetto è Oscar Kenneth Albero che prima ne ha realizzato la versione per il sistema MSX e poi ha convertito il tutto e lo ha infilato in una cartuccia per Sega Megadrive.

Siamo di fronte ad un MetroidVania, quello strano mix tra piattaforme e azione presente nei classici Metroid e Castelvania in cui saltare e sparare è importante tanto quanto esplorare un ambiente aperto e scoprirne i segreti.

Life of Mars ci porta diversi anni nel futuro dove vestiremo la tuta spaziale di Sam, responsabile delle indagini su una colonia marziana che non mostra più segni di vita. Una volta dentro il centro ricerche marziano, scopriremo che c'è più "vita" di quanto si possa pensare... ma è dannatamente ostile!

Il gioco ricorda molto Metroid per lo stile esplorativo, ma ci ho visto molto anche Turrigan per il level design e il tipo di nemici.

Belle le animazioni ma un poco curati i fondali e la palette è forse troppo "spenta". Il sonoro è una riproduzione di quello della versione MSX 2, ma non eccelle e spesso "gracchia".



L'esplorazione è gradevole e coinvolgente nella storia ma in alcuni punti il sistema di controllo del nostro personaggio non è il massimo e spesso ci troveremo a compiere salti "pixel perfect" per poter raggiungere alcune semplici piattaforme.

Avrei spinto di più per quanto riguarda l'esplorazione. È forse più incline al gioco di piattaforme standard che al MetroidVania classico.

La versione MSX 2 mi aveva regalato qualche ora di gioco e il livello di difficoltà era ben calibrato, questa versione Megadrive risente dei comandi non "eccezionali" e in alcuni punti risulta frustrante.

Il titolo è disponibile su sito ufficiale di Kai Magazine al prezzo di 55 euro, forse un tantino alto per questo tipo di gioco.

Adatto agli amanti di questo genere.

di **Giampaolo Moraschi**



GIUDIZIO FINALE



» Giocabilità 75%

Il sistema di controllo non è preciso e spesso l'azione diventa poco godibile.

» Longevità 65%

Non lunghissimo... E 55 euro sono tanti se il titolo si finisce subito.





NEW GAME

BUFONADA

Anno: 2022

Editore: Roolandoo

Genere: Platform/Puzzle

Piattaforma: MSX/MSX2 – Spectrum

Sito web: <https://www.msxdev.org/2022/09/06/msxdev22-16-bufonada/>

Sito web: <https://roolandoo.itch.io/bufonada>

Roolando è il creatore di quel piccolo gioiellino di S.O.L.O altro titolo che partecipò alla MSXDev del 2021 che abbiamo recensito nei numeri 36 ita e 14 uk con dei buoni voti.

È uno sviluppatore che conosce bene il suo lavoro e sforna titoli divertenti e curati.

Questo Bufonada è un'avventura medievale ambientata nel regno di Kimeria, una nazione colpita da una potente maledizione che l'ha gettata in una notte eterna.

Il nostro eroe è Jester il giullare che si trova con l'oneroso compito di capire come far tornare la luce nel suo mondo natale.

Per fare tutto questo si avvale del suo potente animale spirituale... una capra! Spesso ci si aspetta di avere avatar spirituali belli, potenti e maestosi eppure al povero Jester tocca una belante capretta piuttosto malmostosa. Per poterla evocare il giullare dovrà lanciare una piccola sfera di energia et voilà! La capretta sarà a completa disposizione.

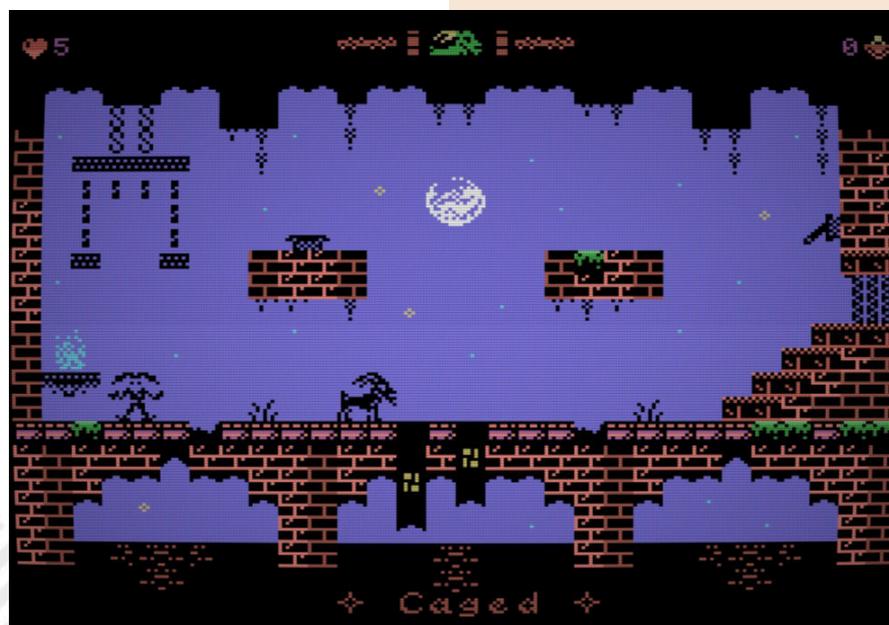
Bufonada è un classico platform a schermo fisso che mette in gioco entrambi gli aspetti del nostro protagonista. Jester può muoversi, saltare e utilizzare oggetti ma non riesce a raggiungere tutte le piattaforme e ha bisogno di sfruttare il suo spirito animale caprino per poterlo fare. Questo comporta la possibilità di switchare durante il gioco tra un personaggio e l'altro. Il tutto con la sola pressione di un tasto. Il quadrupede aiuterà Jester a superare alcuni punti fungendo da trampolino, utilizzerà le sue corna per rompere oggetti e sarà utile per attivare pulsanti.



Un sistema di gioco diverso e divertente e totalmente funzionale. Inizialmente può spiazzare ma funziona e incolla al joystick.

È un titolo geniale e dal game design perfetto.

Graficamente è ben realizzato e dettagliato anche grazie ad una palette "notturna" che crea atmosfera e alle belle animazioni (sia in versione MSX

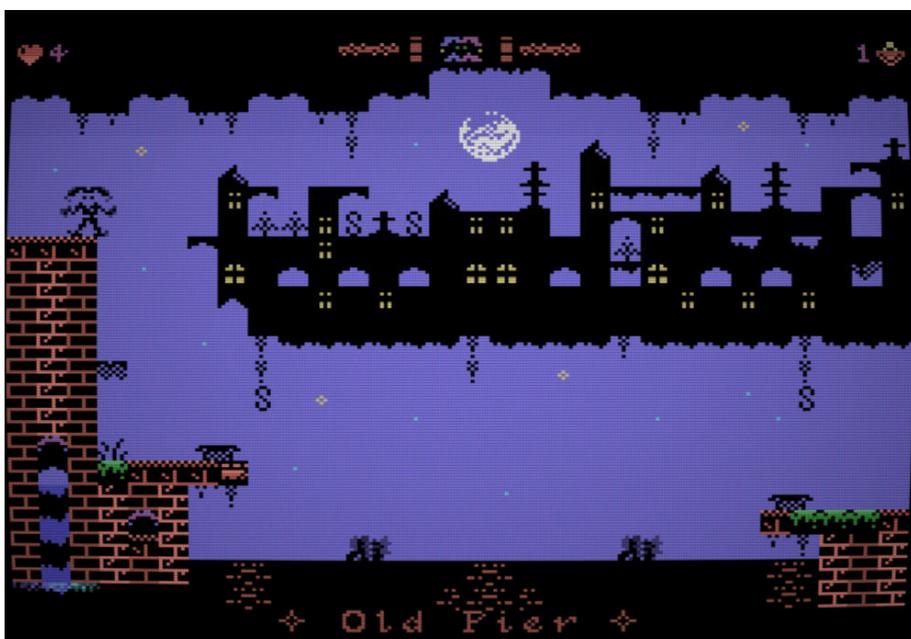




che in versione ZX).
 Comparto audio d'atmosfera e piuttosto sinistro che mi ha evocato alcuni pezzi degli Arcturus (chi li conosce?).
 Non è un gioco facile e spesso ci porterà a ripetere il livello diverse volte. Un po' perché confusi dallo switch un po' perché alcuni livelli sono diabolici però è un grande gioco che ci tiene attaccati allo schermo in entrambe le sue edizioni.

Io ho preferito la versione MSX più fluida ma anche la versione ZX è un bel giocare.
 Consigliatissimo.

di **Giampaolo Moraschi**



GIUDIZIO FINALE

» Giocabilità 90%

Un attento game design che mescola azione, switch dei personaggi e livelli bellissimi.

» Longevità 90%

Inizialmente imprecherete in diverse lingue ma, una volta capito come combinare al meglio il dinamico due, non vedrete l'ora di portarlo a termine.





NEW GAME

MAD STALKER FULL METAL FORTH

Anno: 2022

Editore/Sviluppatore: Strictly Limited – Family Soft

Genere: Beat em up

Piattaforma: Sega Megadrive

Sito Web: [https://](https://www.strictlylimitedgames.com/collections/sega-mega-drive-genesis-partner-store)

www.strictlylimitedgames.com/collections/sega-mega-drive-genesis-partner-store

Mad Stalker: Full Metal Forth è un classico beat em up a scorrimento laterale realizzato da Family Soft/Fill in Café nel 1994 per X68000 di Sharp. Ha avuto un discreto successo in terra nipponica e successivamente è stato convertito e rilasciato su FM Town e Pc Engine CD ROM.

Nel 1997 trovò una bella conversione anche per Playstation 1 e pochi anni fa il team di sviluppo aveva cominciato un porting anche per il 16 bit SEGA. Dopo un paio d'anni eccolo tra le nostre mani.

Il titolo ricorda Kung-Fu Master e noi assumiamo il ruolo di un agente di polizia che prende il controllo del mecha di difesa noto come Hound Dog. Il compito sarà quello di liberare Artemis City dal crimine e sconfiggere un altro mecha chiamato Rising Dog, simile a noi ma votato al crimine.

Il gioco è di natura lineare, il giocatore si muoverà da sinistra verso destra controllando il mecha con il pad e eliminando i numerosi avversari che si paleseranno sullo schermo. Compito facile da scrivere ma difficile da mettere in pratica in quanto ogni avversario richiederà una tattica d'approccio differente.

A fare la differenza è il livello di preparazione al combattimento di Hound Dog. Il nostro robottone presenta uno stile di combattimento degno dei migliori guerrieri di Street



Fighter II, con tanto di mosse speciali e combo hit.

Le aree di gioco da affrontare sono 6 e alla fine di ognuna di queste è presente la più classica delle boss fight. Un titolo che non presenta nulla di nuovo rispetto a molti altri titoli simili ma che sulla carta e, joypad alla mano, risulta molto divertente.





GIUDIZIO FINALE

» Giocabilità 80%

Si lascia giocare bene e i comandi sono ben calibrati. Divertenti le combo moves e i numerosi tipi di attacco.

» Longevità 70%

Sei livelli sembrano tanti o pochi? Dipende da come apprezzate il titolo. Non è eterno ma risulta godibile e spesso lo rigiocherete in una pausa rilassante.



Mi ha sempre ricordato due cose. La prima l'ambientazione alla "Patlabor" (una serie poliziesca anime con robot al posto dei classici agenti di polizia) e due titoli che ho amato particolarmente: Cybernator e Genesis Target Earth.

e godibile. Sicuramente un gioco dove non occorre pensare totalmente ma dove è necessario menare duro e menare forte.

Sufficienza piena.

di **Giampaolo Moraschi**

Questa versione Megadrive risente tecnicamente degli anni. E' identica alla versione per X68000 (leggermente più lenta), i robot sono animati bene mentre fondali e colori non sono nulla di impressionante.

Consiglio di provarlo per capire se può piacere o meno. Io l'ho trovato divertente





NEW GAME

A PIG QUEST

Anno: 2023

Editore/Sviluppatore:

Protovision – Piggy 18

Genere: Platform/Adventure

Piattaforma: Commodore 64

Sito web: [https://](https://piggy18.itch.io/a-pig-quest)

piggy18.itch.io/a-pig-quest

Frank Further è un maiale. Vive a Porkville, una paciosa, scrofolosa e ridente cittadina dove tutti sono felici. Cibo, sole, pozzanghere, mele e letizia sono presenti in questo grazioso paradiso.

Ma nella vita di un maiale non tutto è perfetto... almeno non tutto fino al giorno in cui spariscono le statue del benessere, viene rapita Lady Porchetta e ogni abitante della città si trasforma in un pigro, immobile e incattivito nullafacente.

Il piccolo Frank si lancia nell'avventura (suo malgrado) finendo per errore in una grotta e gettandosi nell'avventura più importante della sua vita.

Si apre in questo modo uno dei titoli più attesi di quest'anno per il Commodore 64, A Pig Quest è forse uno degli esempi di come si possa tirar fuori ancora qualcosa di incredibile sul 8 bit Commodore grazie a passione, conoscenza del sistema e senso della costruzione dei livelli.

E' un platform che ricorda Ghost 'n Goblins ma non solo, sarebbe stato bello ma anche troppo scontato.

Allora il team Piggy 18 ha infilato dentro elementi dei titoli come Zelda o Castelvania trasformandolo in un titolo dal gameplay non lineare dove l'esplorazione la fa da padrone e dove sarà necessario fare attenzione al massimo per sbloccare chiavi nascoste, porte segrete, grimaldelli, botole e trappole.

Ma non basta. Ci sono una marea di citazioni di altri titoli sia visivamente che nel sistema di gioco.

Il piccolo Frank potrà girovagare per i 5 mondi non solo armato di coltellino di precisione e canottiera della salute, ma scovando armi e armature nei



numerosi forzieri presenti (che dovranno essere aperti con una chiave). Ogni arma presenta caratteristiche differenti (il martello di Pork è incredibile) e le armature permetteranno una sopravvivenza maggiore.

I livelli non sono mai banali e presentano varietà incredibili di nemici





e situazioni, ed è il bello di questo titolo, bisogna pensare prima di saltare e blastare e questo ci piace.

Ci sarà anche da scoprire le aree segrete nascoste e la narrazione che spesso ci verrà incontro per capire come approcciarsi ad una determinata situazione/mostro.

Questa meraviglia è stata realizzata dalla collaborazione di Antonio Savona al codice, Mauricet alla grafica e al concept e Aldo Chiummo e Gaetano Chiummo alla musica e sfx.

Chapeau ragazzi! Non c'è storia.

Grafica dettagliata. Un tripudio di pixel art colorata e dalle azzeccate scelte cromatiche. Animata perfettamente e varia, anche troppo. C'è tanto da vedere in A PIG QUEST e il giocatore si perderà ad esplorare i livelli e sviscerarne i segreti pur di voler passare al mondo successivo.

Il SID in questo gioco è stato preso e portato al massimo. I trentasei brani diversi in tantissime ambientazioni

(non solo nei livelli, ma anche nei quadri di gioco) sono realizzati con un amore tale da far emozionare. Un livello musicale impressionante.

C'è tutto e c'è di più. Un sistema di difficoltà calibrato mescolato ad un ottimo controllo via joystick. Ci sono i mostri di fine mondo davvero impressionanti che richiedono determinazione ed un pizzico di esperienza.

I livelli sono strutturati magicamente, disegnati con cura e studiati, pensati, amati... c'è davvero amore in A PIG QUEST.

C'è la passione e la conoscenza.

Mentre scrivo queste parole non è ancora uscita la versione digitale e in box (è l'8 febbraio e il titolo esce il giorno di San Valentino), ma vi consiglio caldamente di correre ad acquistarla. Il titolo è disponibile in formato .crt in versione digitale perfettamente compatibile in real hardware (sia Pal che NTSC) e con i sistemi di emulazione e il THE C64 (maxi e mini).

La versione box sarà in formato cartuccia.

Vale ogni centesimo e non vi pentirete di voler scendere nelle grotte assieme a Frank.

di Carlo Nithaiah Del Mar Pirazzini

GIUDIZIO FINALE



» Giocabilità 95%

Tutto fila perfettamente. Livelli bilanciati, divertenti e ottimi controlli.

» Longevità 95%

Sembrano pochi 5 mondi ma sono 200 schermi da affrontare, studiare e giocare.





NEW GAME

AQUABYSS

Anno: 2022

Editore/Sviluppatore: Aged Code

Genere: Action-adventure, Strategy

Piattaforma: Amiga

Sito web: <https://aquabyss.com/>

Un nuovo titolo per Amiga sontuoso e ambizioso; dal prezzo non proprio "da homebrew" e dalle meccaniche innovative (almeno su Amiga).

Aquabyss dopo anni di sviluppo arriva sul mercato in occasione di Amiga 37 a Monchengladbach in Germania e suscita curiosità-

È un gioco per una macchina retrò ma con richieste dei tempi moderni. L'hardware minimo per eseguire il gioco è un Amiga 68000, 2 mb di chip, 8 di Fast e 15 mb di spazio libero su disco rigido.

Ma, a causa delle particolarità del gioco (come le visite nelle città) è consigliato almeno un 68020 a 28Mhz o, ancora meglio un 68030 a 50 mhz o superiore.

Il gioco attinge molto dalla connessione di rete. Può essere installato localmente attraverso un eseguibile, ma il resto verrà scaricato dal server della Aged Code. Un sistema inusuale su Amiga. Ogni volta che il gioco si attiva e si connette al server verifica la disponibilità di eventuali aggiornamenti. Se presenti, li scarica e li installa automaticamente.

Un'altra caratteristica interessante è l'elenco dei risultati tra i giocatori e la possibilità di interagire con gli altri players attraverso una chat di gioco online.

Il gioco può essere giocato completamente anche scollegato dal web. Spetta al giocatore decidere se connettersi al server o meno. È anche



possibile limitare il gioco online scaricando i soli aggiornamenti.

L'ambientazione di gioco è quella cara ai mondi di Jules Verne e alle tematiche Steampunk.





GIUDIZIO FINALE



» Giocabilità 85%

Un romanzo Steampunk da giocare.

» Longevità 75%

Fondamentale seguire il testo e la storia.

Il gameplay potrebbe risultare ripetitivo.



Gli esseri umani hanno realizzato nelle profondità marine grandi città, simili alle loro versioni di superficie. All'interno di queste città si è sviluppata un'ampia rete di commercio. Lo scopo del gioco è quello di commerciare, scambiare e amministrare la rete commerciale e scoprire anche un piccolo mistero, la sparizione di un nostro parente.

Ci sposteremo tra i sette mari con il nostro fido sottomarino. Una macchina incredibile ma che chiede costante aggiornamenti, modifiche, acquisto di nuovo carburante e manutenzione. L'esplorazione del gioco ci mette di fronte a numerose possibilità. Le città sono tante e sono disposte in numerose zone sottomarine (non tutte raggiungibili a causa della differente profondità), sono diverse le possibilità di interazione, scambio e acquisto ed è possibile esplorare questa affascinante avventura in ogni sua parte.



Aquabys è un gioco che richiede pazienza, voglia di apprendere le meccaniche e tanta profondità. Non è un titolo immediato, richiede ragionamento e una dose di apprendimento elevata.

Consiglio di seguire la modalità storia e di leggere tutto ciò che arriva sullo schermo per apprendere tutte le meccaniche e i dialoghi. Senza ci si bloccherà molto presto.

Se cercate azione lasciate perdere. È un titolo che si ripete e che mette in moto meccaniche già viste in titoli simili per pc come Sid Maier's Pirates ad esempio. Il gameplay non è sicuramente vario e questo potrebbe far storcere il naso a chi cerca qualcosa di diverso. Tecnicamente è bello da vedere e con una bella colonna sonora.

Non è un titolo che costa poco (per i tempi e per Amiga), ma è un'esperienza diversa che mi sento di consigliare e supportare.

di **Roberto Pirazzini**





NEW GAME

MARIA RENARD'S REVENGE

Anno: 2023

Editore/Sviluppatore: Z-team

Genere: Platform

Piattaforma: Amiga 1200/
CD32/A500 mini

Sito Web: <https://z-team.itch.io/maria-renards-revenge>

Maria Renard's Revenge è un titolo realizzato dallo Z-team, un gruppo che ama particolarmente la saga Castelvania di Konami. Titolo sviluppato con l'ausilio dello Scorpion Engine, motore di sviluppo che sta facendo tanto bene alla scena Amiga negli ultimi anni.

Ma chi è Maria Renard? E' un personaggio della saga di Konami. Una giovane cacciatrice di vampiri che vanta una lontana parentela con la famiglia/clan dei Belmont.

Aveva solo 12 anni quando apparve per la prima volta nella serie, ma era già dotata di una potente conoscenza della magia, degli spiriti famigli e della magia delle bestie celestiali.

Il primo titolo dove appare è

Castelvania: Rondò of Blood dove si scontrerà con Dracula in persona.

Si tratta di un personaggio piuttosto amato nella serie che cambia il modo di approcciarsi nei numerosi titoli in cui appare.

In questo fan game per Amiga, la nostra eroina dovrà attraversare 4 livelli pieni di mostri e trappole e raggiungere ... il male che infesta il suo mondo.

Il titolo è stato realizzato partendo dalle mappe grafiche e gli sprite dei titoli Konami e presente una gradevole e pixelosa grafica.

Tutto è molto colorato e gradevole e le numerose animazioni sono piacevoli come è piacevole la colonna sonora di accompagnamento.





GIUDIZIO FINALE

» Giocabilità 60%

Alcuni problemi di programmazione compromettono il risultato e la giocabilità.

» Longevità 60%

Da fan per i gli appassionati di Castelvania.

La difficoltà è sicuramente da riguardare.

Il gioco è scaricabile gratuitamente dal sito dello sviluppatore ed è disponibile in diversi formati.

La versione per A1200/CD32 (dove basta 1Mb di Fast), la versione per A500 mini e una versione per Windows che lancia la rom in modalità emulazione. Arriviamo a tirare le somme.

È bello ma non balla si dice in questo caso. Graficamente, come dicevo, gradevole ma mancano alcune cosette (che spero vivamente verranno corrette in future release). Sono presenti numerosi glitch grafici che non fanno comprendere bene alcune piattaforme. Il sistema di gioco non è fluidissimo e spesso si muore senza aver capito cosa ci ha colpito.

A questo aggiungiamo uno strano bilanciamento della difficoltà che vede il primo livello molto "tenace" e i successivi piuttosto "lisci".

Insomma è un gioco che si è fatto attendere dalla comunità ma ha difetti evidenti.

Peccato.

di Carlo Nithaiah Del Mar Pirazzini





NEW GAME

S.A.K.

Anno: 2023

Editore: SAK team/Future Disk

Genere: Action

Piattaforma: MSX2

Sito web: <http://futedisk.nl/fd47.php>

Green Beret (Rush n Attack) è stato uno dei titoli da sala giochi più amati e allo stesso tempo più odiati di sempre.



Uscito nel 1985 da Konami era ricordato per la sua ambientazione in stile "guerra fredda" e la dipendenza assoluta che il giocatore aveva nel dover utilizzare un coltello per sopravvivere ai nemici. Un titolo decisamente difficile ma che creava dipendenza. Tutti volevano vedere cosa c'era nel quadro successivo.

Questo S.A.K per MSX 2 è una conversione del gioco classico, anzi direi un omaggio del titolo Konami. Quattro livelli di gioco è un grado di



sfida impressionante, il tutto accompagnato da una grafica fluida e colorata e da un sonoro funzionale. Il gioco è sempre quello e affrontare i "terribili sovietici" non è una passeggiata.



Un titolo adatto agli appassionati di Green Beret e agli hardcore gamer che cercano una sfida degna di Ercole. di **Giampaolo Moraschi**



GIUDIZIO FINALE

» Giocabilità 70%

Si gioca bene con il pad/joystick e il titolo fila abbastanza liscio.

» Longevità 65%

Come il titolo originale vi farà imprecare non poche volte.





MYSTIC PILLARS

Anno: 2008

Sviluppatore: Sivak Games

Genere: Puzzle Game

Piattaforma: Nintendo NES

Sito web: [https://](https://www.retrousb.com/product_info.php?cPath=23&products_id=55)

[www.retrousb.com/](https://www.retrousb.com/product_info.php?cPath=23&products_id=55)

[product_info.php?](https://www.retrousb.com/product_info.php?cPath=23&products_id=55)

[cPath=23&products_id=55](https://www.retrousb.com/product_info.php?cPath=23&products_id=55)



Ispirato al gioco Columns e al mitico Puyo Puyo, Mystic Pillars segna il terzo titolo su NES da parte dei Sivak Games. Lo scopo è quello di formare con le tessere che scendono dall'alto una riga, colonna o diagonale di 3 o più tessere corrispondenti. In questo modo cancelleremo quelle tessere e tutte quelle sopraelevate cadranno. Più andremo avanti nel titolo e più il gioco aumenterà livello di difficoltà

e velocità. Ogni tanto verranno rilasciate tessere speciali che elimineranno tutte le tessere del colore dove si appoggeranno.

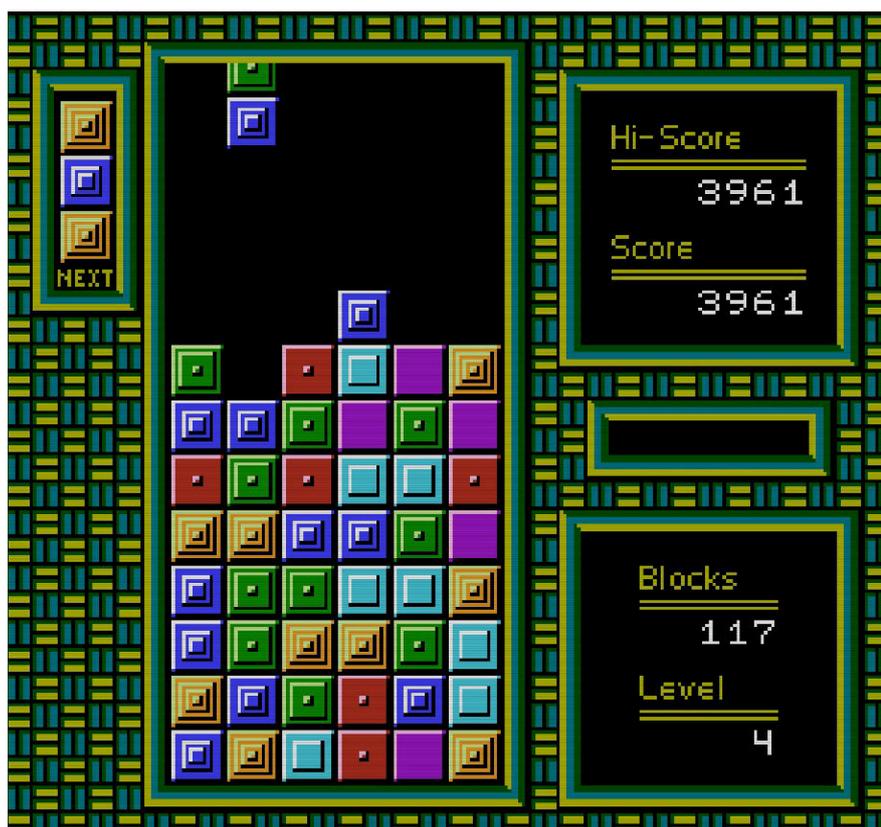
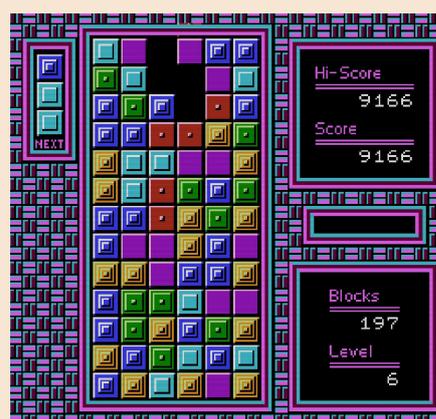
Sono disponibili 5 livelli, 3 brani musicali e 8 sfondi personalizzabili. Mystic Pillars doveva essere rilasciato nel 2008, ma è stato posticipato a causa di numerose problematiche per reperire il materiale per realizzare le cartucce.

È un titolo semplice e simpatico che non brilla certo per originalità ma che risulta gradevole da giocare e super semplice da apprendere.

Il livello di difficoltà è ben bilanciato e questa cosa lo rende gradevole.

Ci sono titoli sicuramente superiori ma Mystic Pillars si merita la sufficienza piena.

di **Giampaolo Moraschi**



GIUDIZIO FINALE

» Giocabilità 65%

Tutto già visto ma funziona bene. Sufficiente

» Longevità 70%

Difficoltà ben bilanciata.





THE FIREMEN

Anno: 1994

Editore: Human Entertainment

Genere: Action

Piattaforma: Super Nintendo

Anno 2010, una nota azienda chimica sta organizzando la sua festa di Natale annuale nei locali.

Scoppia un terribile incendio in cucina e, in poco tempo, tutto l'edificio è in fiamme alimentato dai numerosi prodotti chimici al suo interno.

È il momento di entrare in gioco! La squadra dei vigili del fuoco Pete (noi), Danny, Max, Walter e Winona rinomati per il loro coraggio e le loro abilità si gettano nell'avventura per salvare gli sfortunati dipendenti e i loro familiari. Certo che peggior giorno di Natale proprio non si poteva avere vero?

Fireman è un titolo che zio Nith mi ha dato da provare e devo dire che mi sono divertita tantissimo.

La cosa più piacevole di The Firemen è proprio la rapidità con cui l'azione

- o meglio, le fiamme - ci viene addosso mentre si va da una stanza all'altra, spegnendole tutte. In sostanza, il gioco si svolge come una corsa e la pistola incontra l'ibrido dell'inferno di proiettili, con la manichetta antincendio come unica difesa che abbiamo per non essere bruciati fino a diventare croccanti.

Possiamo usare il tubo in un paio di modi diversi: puntandolo in avanti per sparare un getto nella direzione in cui siamo rivolti per spegnere le fiamme a distanza o puntandolo verso il suolo per affrontare gli incendi in rapido movimento lungo il pavimento. Ci sono molti modi originali in cui l'inferno può minacciare anche noi.

Ci sono fiamme che si diffondono rapidamente, fiamme che esplodono





GIUDIZIO FINALE

» Giocabilità 80%

Meccaniche interessanti e bei livelli da esplorare.

» Longevità 70%

Sei livelli impegnativi ma alla fine non lo rigiocherete tanto spesso.

dai tubi e persino le palle di fuoco proiettili che metteranno a dura prova i nostri riflessi. Come se non bastasse, ci sono anche fragorose esplosioni da affrontare, il pavimento stesso che crolla e robot di servizio fuori controllo... e questi sono solo alcuni dei primi livelli! The Firemen presenta solo sei livelli in totale. Le cose sono ragionevolmente gestibili in termini di difficoltà, ma i livelli successivi richiedono molta precisione e agilità, con gli incendi che si diffondono a un ritmo sempre più veloce e i danni che ne derivano diventano sempre più letali. Per fortuna, i controlli sono abbastanza reattivi da garantire che dipenda davvero dalle nostre abilità. Il ritmo di gioco in The Firemen è spesso furioso, ma coinvolgente, il che è fondamentale per rendere questo gioco un'esperienza divertente.

Completare i sei livelli del gioco non è un'impresa da poco, la rigiocabilità del titolo una volta terminato è piuttosto limitata. Alla fine di ogni livello, ci verrà assegnato un punteggio che descrive in dettaglio la quantità di fuoco che abbiamo spento per il determinato piano in cui ti trovi. Quindi per i completisti c'è una sfida da affrontare. Ma tornare di nuovo nel gioco rivela gli stessi focolai di incendio e gli stessi schemi di livello. Questo è un pensiero del tutto pio (e nerd) da parte mia, ma una generazione procedurale casuale del fuoco avrebbe potuto essere un

modo eccellente per aggiungere un po' di rigiocabilità. Diavolo, anche un vero editor di livelli con livelli salvati su SRAM sarebbe stato fantastico, ma sono sicura che è andato ben oltre le intenzioni di Human con il gioco.

Tuttavia, The Firemen racchiude abbastanza elementi perché valga la pena giocarlo. La narrativa in stile film d'azione americano, oltre alla colonna sonora veloce che l'accompagna, mantiene le cose avvincenti e la grafica spensierata in stile anime gli impedisce di prendere le cose troppo sul serio.

Ora basta, torno a spegnere incendi e manichette... l'esame di chimica può attendere.

di **Ingrid Poggiali**





NEW GAME

MINKY

Anno: 2022

Editore/Sviluppatore:

Matze1887

Genere: Platform

Piattaforma: Amiga

Sito web: [https://](https://matze1887.itch.io/minky)

matze1887.itch.io/minky

Minky è un nuovo platform per Amiga uscito a fine 2022. È un titolo piacevole e divertente che ci vede nei panni della piccola scimmietta Minky che deve superare 30 livelli e recuperare tutte le monetine presenti.

Un gioco semplice, lineare e che appartiene alla “vecchia scuola” dei giochi di piattaforme.

Si tratta di un titolo piacevole da giocare e divertente, abbastanza impegnativo alla fine ma mai da diventare impossibile o frustrante.

Il sistema di password per permettere di proseguire è fatto bene ed i livelli sono abbastanza vari e godibili.

Mi ha ricordato lo stile di giochi Night Owl Design, il sound è ben fatto e si ascolta con grande piacere con alcune melodie molto piacevoli.

L'unica pecca che ho riscontrato è nella gestione dei salti. Non poche saranno le volte in cui imprecherete come portuali per raggiungere la punta più estrema di una piattaforma. Graficamente è molto gradevole nelle scelte cromatiche e nelle animazioni,



un prodotto davvero ben fatto.

È possibile giocare a Minky in tutti i modi possibili: emulazione, A500 mini e su real hardware (le richieste di gioco sono minime, 1 mb di Ram).

Forse non sarà il titolo più originale mai uscito su Amiga ma è un buon platform.

di **Giampaolo Moraschi**



GIUDIZIO FINALE

» Giocabilità 70%

Bei livelli ma il sistema di salto “pixel perfect” è micidiale.

» Longevità 85%

30 livelli sono tanti da affrontare e il titolo si fa giocare volentieri.





SHADOW WARRIORS NINJA GAIDEN



Ninja's Skyscraper Fight/Ninja Ryūkenden GB Matenrō Kessen/Shadow Warriors Ninja Gaiden/Ninja Gaiden Shadow... Wow! Mille nomi un solo gioco!

Ciao! Come state? Spero bene... Torno con una chicca tutta cinese da mostrarvi, il bellissimo Shadow Warriors! Poco apprezzato dai fan della serie del nostro Ninja, a mio dire è un gioco davvero pazzesco se mettiamo in conto che è su Gb!

Il nostro protagonista Ryu Hayabusa ne dovrà affrontare parecchie, ma tiene duro per 5 stage che terminano con un boss.

Correva l'anno 1991 e, in Giappone, il team Tecmo se ne uscì dicendo: "Ma se facessimo un Ninja Gaiden per GB vi piacerebbe?". Tanto fecero e tanto tanto dissero che, il 13 settembre dello stesso anno, il progetto giunse nei negozi. I fan di Ninja Gaiden generalmente lo snobbano ma non è un buon motivo per non provarlo, visto che entra di diritto



nella categoria "Titolo che non ti aspetti", in senso positivo ovviamente. Il motivo per cui Shadow Warriors non è particolarmente apprezzato è da collegare al fatto che sembra una versione modificata di "Shadow of the Ninja", un (noto?) gioco della Natsume.

Personalmente io volevo fortemente averlo perché ha una grafica davvero pazzesca, con sprite ben definiti ed una notevole fluidità, cosa non comune per un gioco GB. Il nostro personaggio si muove con grande agilità per i livelli ed è un piacere controllarlo. Non mancano vari power up, tra cui la fune ninja "Ninchaku", dal funzionamento simile al caro braccio meccanico di "Bionic Commando". Non useremo Shuriken, ma avremo una "katana spacca tutto". In base ai potenziamenti raccolti, potremo inoltre usare limitatamente la "Ruota dell'arte del fuoco" che ripulirà lo schermo dai nemici.

Attraverseremo cinque piacevoli stage affrontando Ninja, Robot, Robot volanti, Raggi laser da evitare, facendo tutto a pezzi con la nostra spada. Una delle poche pecche sono i boss, davvero facili e mutevoli (come Madonna ai tempi d'oro).

Questi saranno spesso camaleontici e discretamente improbabili: volete un esempio? Il boss finale che inizia come Dracula e dopo un po' diventa un robottino alla Mazinga Z!

Tecnicamente godibile e dalla buona giocabilità.

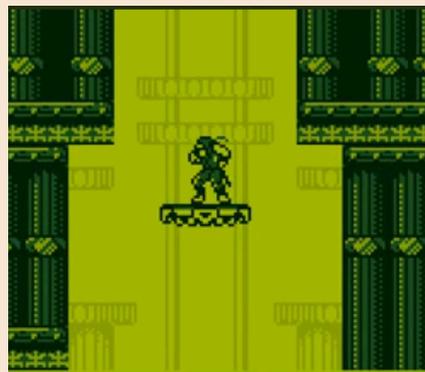
di **Barbara "Morgana" Murgida**

Anno: 1991

Editore: Tecmo

Genere: Beat em up

Piattaforma: Game Boy



GIUDIZIO FINALE

» Giocabilità 75%

Un buon design dei livelli e tanta roba da distruggere e malmenare.

» Longevità 75%

Non un titolo semplice ma godibilissimo.





CRYSTALIS



Crystalis è un action rpg del 1990 prodotto da SNK per il NES.

Il gioco inizia con un giovane mago che si risveglia da un sonno criogenico in cui era stato imprigionato durante la Grande Guerra e prima della terribile apocalisse nucleare scatenata sul mondo nel 1997. Non ricorda nulla di sé stesso o su chi fosse precedentemente ma viene coinvolto in una grande avventura che lo porterà al salvataggio del mondo.

Il titolo è realizzato come molti classici del periodo, con una visuale dall'alto verso il basso. Il pad controlla il movimento attraverso la croce direzionale, il pulsante d'azione viene utilizzato per combattere con una spada mentre il pulsante secondario per poter utilizzare un potere magico o un oggetto richiamato dall'inventario.

È possibile equipaggiare il nostro eroe con varie armature, oggetti magici e scudi. Sconfiggere i mostri e risolvere le quest permette di guadagnare punti esperienza e aumentare di conseguenza il livello del



nostro personaggi (Hp, difesa, attacco, ecc). Il mezzo principale per sconfiggere i mostri è attraverso l'uso di quattro spade elementali sparse nel gioco (la quinta spada chiamata Crystalis, si ottiene solo nel mondo finale). Queste armi possono colpire sia con normali fendenti che con attacchi potenziati dagli elementali contenuti al loro interno. Questa loro natura elementare fornisce un mezzo necessario ed efficace per affrontare determinate situazioni.

Alla sua uscita il titolo fu elogiato per la sua resa grafica avanzata che spingeva il NES al suo massimo. Stiamo parlando del periodo di fine vita della console e gli sviluppatori conoscevano molto bene come lavorare sul 8 bit di Nintendo. Ricorda lo stile di Nausicaa nella Valle del vento. Molto colorata, animata bene e piuttosto varia che si accompagna perfettamente ad una colonna sonora impressionante e coinvolgente. Senza dubbio la miglior colonna sonora mai realizzata su un gioco Nintendo 8 Bit.

Tutt'ora a livello scenico è un titolo appagante ma presenta diversi problemi in termini di gioco.

L'azione è spesso ripetitiva, diversi nemici si affrontano tutti allo stesso modo premendo continuamente e rapidamente il pulsante.

Problema di ripetitività negli scontri a parte, il titolo è senza dubbio un prodotto di qualità e con una trama profonda e ben scritta.

Forse uno dei migliori ARPG per NES. Da riscoprire e giocare.

di Carlo Nithaiah Del Mar Pirazzini

Anno: 1990

Editore/Sviluppatore: SNK – Digital Eclipse

Genere: Action RPG

Piattaforma: Nintendo NES/ Game Boy Color



GIUDIZIO FINALE

» **Giocabilità 85%**

Trama profonda e coinvolgente.

» **Longevità 75%**

La ripetitività degli scontri non aiuta e annoia.



JAMES BOND 007 LIVE AND LET DIE

Anno: 1988

Editore: Domark

Genere: Azione

Piattaforma: Commodore 64

Si sente già la primavera alle porte, le giornate tendono ad allungarsi e purtroppo o per fortuna, quest'anno Milano è rimasta a digiuno della neve. Spero che questo nuovo anno sia iniziato molto bene per tutti, dopo due terribili anni di restrizioni in lotta con un nemico invisibile ma temibile. Comunque cari fan di 007 e Buggy Boy ho il piacere di parlarvi di Live and Let Die, uno dei tanti giochi tratti dalla saga cinematografica dell'agente segreto James Bond 007 che dagli anni 60 ci regala spettacolo, intrighi, colpi di scena grazie al fascino del cast molto bravo, oltre naturalmente a numerosi titoli per computer e console. Questo gioco targato Domark e sviluppato da Elite Systems è un insieme di gioco di corsa e azione con protagonista un motoscafo con visuale da dietro pilotato da noi e da James Bond. Lo stile di gioco ricorda molto Buggy Boy che quasi tutti ormai conoscono come le proprie tasche con molte similitudini come ad esempio tronchi per saltare, rampe, gallerie e qualche piccola differenza: la corsa si svolge su acqua anziché su pista, la mancanza di checkpoint e al posto del timer con conto alla rovescia, un il serbatoio del carburante che piano piano si svuota come il nostro conto in banca quando siamo a Courmayeur in piena stagione. Potremo sparare con munizioni infinite più tre missili per i nemici più corazzati e persino abbattere le pareti.

I percorsi sono quattro e ciascuno vi terrà impegnati per una decina di minuti abbondanti. Si inizia con il percorso di pratica in cui sono presenti solo scogli e bersagli rossi e neri. I primi dovremo centrarli con lo sparo ordinario, mentre i secondi con i missili; lo scopo del percorso è piuttosto semplice, ossia centrare tutti i bersagli. Il secondo ed il terzo livello sono anch'essi delle esercitazioni ambientate in Antartide e nel deserto del Sahara, ma questa volta sono presenti molti più grattacapi come motoscafi

nemici, mine, aerei che sganciano bombe, torrette mobili ecc. Il quarto e ultimo tracciato è la missione vera e propria: a New Orleans dovremo arrivare fino alla raffineria di Mr Big (antagonista presente anche nel film) per farla saltare. Se volete sapere già adesso come, vi dico soltanto, non sprecate i missili perché anche se di tanto in tanto passerà un elicottero che ve li fornirà, molte volte invece tirerà per la propria strada senza nemmeno farvi un cenno di saluto.

A livello di giocabilità non è nulla di che una volta che si prende la mano. Discorso a parte lo merita lo sparo: molte volte ho trovato difficile colpire i nemici soprattutto in diagonale preferendo piuttosto evitarli. Il sonoro ricalca la colonna sonora del film ma solo nella schermata iniziale, nel resto del gioco si sentiranno solo il rombo del motoscafo e il rumore degli spari.

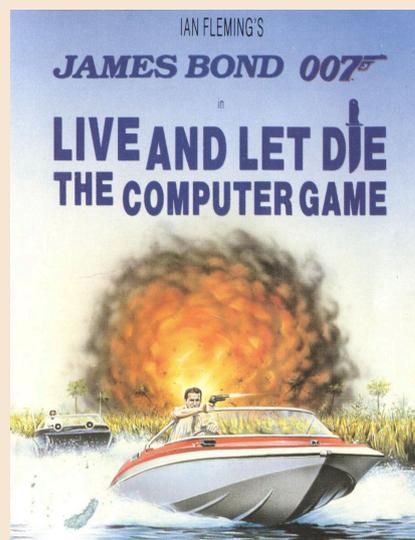
L'unico modo per far terminare una partita è rimanere con il serbatoio a secco. E qui sento in dovere di darvi un consiglio; andate sempre alla velocità massima e non trascurate le taniche di carburante che troverete lungo il percorso, contraddistinte da questi colori: rosso per quantità minima, verde per media e grigio per quasi il pieno.

A livello di longevità quattro percorsi potrebbero sembrare pochi, ma vedrete che saranno sufficienti viste le difficoltà ed il tempo che si impiega per arrivare alla raffineria.

Nel gioco non esistono salvataggi, continue, crediti, perché non segue una storia lineare. Una volta terminato un percorso la partita finirà e deciderete voi l'ordine con cui giocare: tutti i percorsi di fila o uno alla volta.

Buon inizio primavera e divertitevi con questo titolo di una leggendaria saga!

di **Daniele Brahimi**



GIUDIZIO FINALE

» Giocabilità 80%

Sparo un po' ballerino ma comunque piacevole.

» Longevità 75%

Quattro livelli, non molti ma parecchio tosti!



C'è stato un momento, qualche anno fa, in cui interagendo con amici e colleghi che ricordavano i tempi della nostra giovinezza ho iniziato a chiedermi se indugiare in certi ricordi non fosse altro che un poco salutare rigurgito di infanzia.

Tutti questi adulti che parlavano di cartoni animati, cantandone le sigle, o di piattaforme quasi scomparse, era da un lato commovente e dall'altro un po' triste. Probabilmente con un pizzico di arroganza, mi dicevo che nel mio caso c'era qualcosa di più oltre le apparenze, senza sapere però bene cosa. In realtà qualcosa di più c'era veramente, e cioè la voglia non solo di ricordare ma anche di fare, di recuperare il tempo "perduto" e portare la nostalgia ad un nuovo "livello", trasformando una semplice struggente sensazione in una potente e moderna linea di azione. Non era certamente un tentativo di essere migliore di altri (non lo sono assolutamente), ma solo quello dare concretezza a qualcosa che rischiava di andare perduto.

Come molti lettori ormai sapranno, la mia "carriera" videoludica è iniziata nel 1982 con il VIC-20 di un amico ed il primo gioco è stato "The Count", avventura testuale di Scott Adams. Da allora le text adventures sono rimaste nel mio cuore e, anche se in realtà ne ho giocate molto poche (alcune erano ben difficili, l'inglese era ancora una "lingua aliena" e le soluzioni non si trovavano), esse fanno tutt'oggi parte di uno dei miei generi preferiti.

Qualche anno dopo comprai il mio primo computer, il C128, macchina tanto bella quanto incompresa e pertanto snobbata. Il software disponibile era scarso ed i giochi inesistenti (a meno di passare in modalità C64, ma quella è un'altra storia). Sono stato costretto a scrivere da solo tutto i programmi di cui avevo bisogno, cosa senz'altro molto formativa, ma a volte i risultati sono stati scarsi. In particolare, le poche text adventures che ho scritto erano poco di più che una demo tecnica, per nulla presentabili all'esterno della mia cameretta.

Ebbene, che centra tutto questo con le considerazioni di apertura?

Beh, tanto per iniziare, il punto focale è che non bisogna mai demordere e perdere la speranza; in secondo luogo, che c'è sempre tempo per passare al nuovo "livello"; terzo, che nostalgia ed infantilismo sono solo negli occhi di chi guarda: come diceva Bruce Wayne, "Sono le mie azioni che mi qualificano."

Il mese scorso ho preso la palla al balzo e mi sono iscritto alla PunyJam#3, una competizione tra appassionati di IF (Interactive Fiction) che si propone di far conoscere il linguaggio "Inform" (quello creato dalla Infocom per sviluppare tutti i suoi giochi, a partire da "Zork") al popolo dei coders. Pur non conoscendo Inform, mi sono buttato nella mischia, ne ho imparato i rudimenti e ho scritto una text adventure completa. Per quanto possa essere considerata "basica" e tutto sommato lineare, è perfettamente funzionante e gli strumenti moderni mi hanno permesso di compilarla a giuocarla su miei 8-bit preferiti, a cominciare dal mio fido C128.

Pochi giorni fa ho compiuto 53 anni ed un software un tempo irrealizzabile è ora assolutamente reale e pronto ad essere condiviso con la comunità. Altro che nostalgia ed infanzia; il futuro non è mai stato più roseo e più "retro" di così.

Gianluca Girelli

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 7 - Numero 42 - FEBBRAIO 2023

Direttore Responsabile

Francesco Fiorentini

Vice Direttore

Marco Pistorio

Coordinatore Redazione/Editing

David La Monaca

Responsabili Area Web

Giorgio Balestrieri

