



RetroMagazine

future days are back

World



NUMERO 35

FEBBRAIO 2022

- WWW.RETROMAGAZINE.NET

- PUBBLICAZIONE GRATUITA

TANTISSIME

RECENSIONI



Konami su AMIGA!!!



**HARDWARE REVIEW
FAIRCHILD CHANNEL F**

Hardware: ANALOGUE POCKET: il portatile definitivo - GAME BOY: le (S)protezioni
Rubriche: R-TYPE Shoot em up ANNO ZERO - UN BIT DI RARITA': CREATING Q*BERT
Programmazione: LINGUAGGIO FOCAL prima parte - FRATTALI IN BASIC terza parte
RETRO PROGRAMMING ITALIA: Programmiamo in Basic - Musica Maestro SID
Ricordando JERRY LAWSON
... e tanto altro ancora!!!

La fame aguzza l'ingegno

RetroMagazine World, ad ogni uscita nelle due versioni, in italiano e in inglese, registra ormai un sempre crescente numero di lettori e suscita un costante interesse per gli argomenti trattati. La nostra scelta di evitare la distribuzione di



copie stampate, oltre a tenerci lontano da una serie di problemi derivanti dalla gestione degli abbonamenti e delle spedizioni, porta alcuni vantaggi, non ultimo quello di continuare a garantire la completa gratuità della nostra iniziativa editoriale, e ci permette di arrivare ovunque nel mondo con un semplice URL. Il numero di download dei nuovi numeri così come di quelli precedenti testimonia la bontà della decisione finora adottata. Peraltro la licenza Common Creative della rivista non vieta di stampare copie cartacee e, in effetti, siamo a conoscenza del fatto che molte associazioni e gruppi organizzati di retrocomputing producono copie rilegate della rivista e la mettono a disposizione a tutti i membri dei loro club.

Un'altra differenza di RMW rispetto alle "classiche" riviste dei favolosi anni 80 e 90 è la mancanza di una rubrica permanente dedicata alla Posta dei Lettori. Una delle ragioni più importanti per cui esiste RMW è la preservazione della conoscenza e della memoria di tutti i sistemi di home computing della storia, per cui diamo molta importanza al feedback e all'interazione con tutti gli appassionati vecchi e nuovi di retrocomputing. Ma lo scenario (rispetto ai primi anni di diffusione delle macchine a 8/16 bit) è totalmente cambiato. In passato, le riviste in edicola erano praticamente l'unico mezzo dinamico su cui viaggiavano le novità e le redazioni erano per forza di cose il punto di riferimento di tutti coloro che si avvicinavano per la prima volta al mondo dei computer e delle console da gioco. Spesso le rubriche della posta venivano bersagliate da domande di ogni genere e in alcuni casi, più che servire come centro d'informazione generalizzata, diventavano terreno di scontro fra "tifosi" delle varie piattaforme.

Al giorno d'oggi disponiamo di decine di possibili strumenti d'interazione (dalla semplice e-mail fino ai più evoluti tool di instant messaging) ma il feedback che riceviamo dai lettori riguardo a discussioni e approfondimenti suscitati da articoli e recensioni che appaiono nella rivista, è abbastanza esiguo. Noi di RMW gestiamo una pagina Facebook, un sito web, account Twitter, Instagram, Telegram, YouTube, ecc. e riteniamo un obbligo per noi aprire e gestire il flusso delle informazioni in tutti questi canali verso chiunque voglia interagire, ma, a parte qualche caso isolato, i commenti dei lettori si concentrano spesso attorno a qualche post pubblicato sulla pagina FB. Troppo poco, soprattutto se pensiamo che il retrocomputing negli ultimi 4-5 anni ha contagiato, perdonate l'uso del termine, centinaia di migliaia di appassionati anche appartenenti alle giovani generazioni.

E allora, cos'è cambiato per i lettori quando leggono una rivista come RMW, seppure pubblicata soltanto in digitale? Non ci saremmo mica impoveriti noi stessi nella nostra capacità d'interazione? Siamo forse diventati più pigri e più inclini a "subire" la gran messe d'informazioni che la Rete ci propina in continuazione in forma di pillole? Preferiamo oggi un tipo di informazione meno strutturata e rigorosa e più basata sul passaparola e sull'esperienza dei singoli veicolata sui social media e sui forum digitali? Non sarà che 40 anni fa la mancanza di strumenti e di documentazione acuiva il nostro desiderio d'informazione e di scambio comunicativo con i vari interlocutori? Non sarà che soltanto la "fame" spinge all'inventiva e aiuta la creatività e la discussione?

David La Monaca

SOMMARIO

◇ GameBoy e GameBoy Color - le (s)protezioni	Pag. 3
◇ ZX99 e altri add-on non ufficiali...	Pag. 8
◇ Analogue Pocket...	Pag. 10
◇ Fairchild Channel F	Pag. 12
◇ Il linguaggio FOCAL sul PDP-8 - parte 1	Pag. 18
◇ Frattali in BASIC in salsa 8-bit - Parte 3	Pag. 22
◇ C128 in modalità 80 colonne - parte 2	Pag. 26
◇ Come creare un gioco in BASIC per C64 - pt. 1	Pag. 28
◇ Musica Maestro SID!	Pag. 32
◇ Intervista a Bas Scheijde...	Pag. 35
◇ Intervista ad Alessandro Rogati...	Pag. 38
◇ Ricordando Jerry Lawson...	Pag. 42
◇ Warren Davis e l'invenzione di Q*bert	Pag. 44
◇ A volte ritornano...	Pag. 45
◇ R-TYPE - Shoot'em up anno zero	Pag. 46
◇ The Age of Heroes (C64)	Pag. 50
◇ The Shadows of Sergoth (Amiga)	Pag. 52
◇ Puzzle Bobble (C64)	Pag. 54
◇ Wizard of the Board (N64)	Pag. 56
◇ Silver Valley (SMS)	Pag. 58
◇ Nightmare (Amiga)	Pag. 60
◇ Retaliante C.E. (C64)	Pag. 62
◇ Mega Turrigan Director's Cut (MegaDrive)	Pag. 64
◇ Eyra, the Crow Maiden (MegaDrive)	Pag. 66
◇ Knight Guy in Low Res World (Atari 7800)	Pag. 68
◇ Robot in Action (C64)	Pag. 69
◇ Shonen Jump's One Piece (GameBoy)	Pag. 70
◇ Gremlins 2 the New Batch (GameBoy)	Pag. 71
◇ Crisis Force (NES)	Pag. 72
◇ Classic Kong (SNES)	Pag. 74
◇ Prohibition (C64)	Pag. 75
◇ Retro Multiplayer	Pag. 76

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

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





GameBoy e GameBoy Color - le (s)protezioni

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

Canale youtube: <https://www.youtube.com/channel/UCEw0CQ8LKya9jVvWXkEwp4Q>

Chi non sa cosa sia un Game Boy alzi la mano. Nessuno? Bene, ne ero sicuro! Così come sono sicuro che anche il Game Boy Color sia nella memoria di tutti.

Non tutti però sanno che il suo nome in codice era DMG-01 (1989-1990), mentre quello del Game Boy Color (1998) era CGB-01.

In ogni caso le prime "piccole" e portatili di casa Nintendo, sono di sicuro tra le più famose perle mai uscite nel mondo delle console ed anche loro non sono da meno rispetto alle "sorellone", riguardo i sistemi di protezione.

Il Game Boy è infatti dotato di una BootROM che ritroviamo anche nel Game Boy Color (GBC) e nel Super Game Boy (cartuccia-adattatore per giocare ai giochi GB su Super Nintendo).



LA PROTEZIONE DELLE PRIME PORTATILI NINTENDO

La BootROM di cui è dotato l'hardware di GB, GBC e SGB è di 256bytes ed ha 3 particolari caratteristiche "protettive":

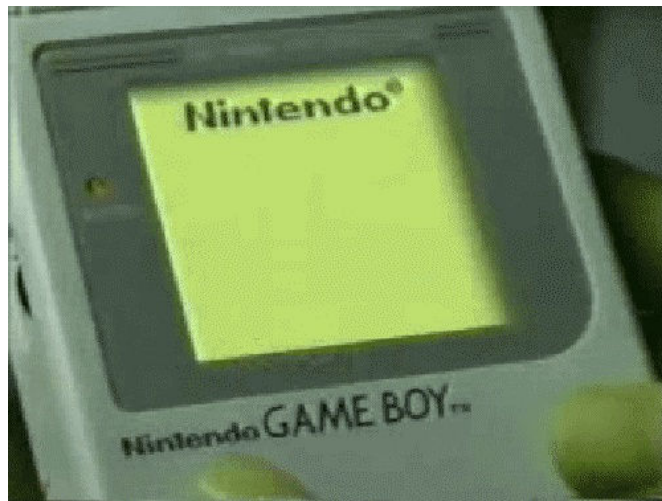
1 – La stessa BootROM viene esclusa dalla lettura con l'ultima istruzione del suo codice: questo "lock out" forse non è stato voluto, ma si è probabilmente reso necessario per motivi legati alle (scarse) risorse di sistema. La BootROM viene mappata in memoria tra l'offset 0x0000 e l'offset 0x00FF (256bytes). Gli ultimi 2 bytes rappresentano un'istruzione che scrive in un registro speciale bloccando tale indirizzo di memoria dalla lettura, in modo tale che tutte le istruzioni successive (che sono, a partire dall'offset 0x0100, quelle contenute nella cartuccia) non possano leggerla;

2 - al suo interno contiene del codice che legge i 48 bytes del logo NINTENDO presenti nella cartuccia (offset che

vanno da 0x104 a 0x133 di ogni cartuccia):

```
CE ED 66 66 CC 0D 00 0B 03 73 00 83 00 0C 00 0D
00 08 11 1F 88 89 00 0E DC CC 6E E6 DD DD D9 99
BB BB 67 63 6E 0E EC CC DD DC 99 9F BB B9 33 3E
```

e li mostra a schermo facendo scorrere dall'alto in basso il notissimo logo Nintendo:



dopodiché li confronta con quelli memorizzati nella BootROM e se sono diversi la console si blocca.

Il fatto di leggere il logo direttamente dalla cartuccia spiega il perché, se accendete un Game Boy senza inserire alcun gioco, appare questo rettangolo nero (i bytes del logo sono tutti ad FF):



Per una descrizione completa di come funzionano i byte del logo vi rimando a questo meraviglioso articolo: <https://catskull.net/gameboy-boot-screen-logo.html>





3 - un secondo controllo effettuato dopo quello del logo è quello di sommare i valori che vanno dall'offset 0x0134 a 0x014C ed aggiungere poi il valore presente in 0x014D: se la somma è zero il programma continua altrimenti la console si blocca.

Ecco uno screenshot di una ROM GB a caso dove potete verificare gli offset menzionati:

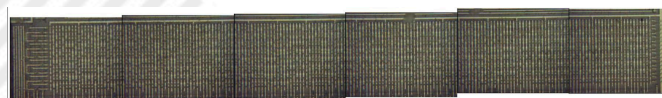
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	83	5F	3E	00	8A	57	C9	FF	86	22	3E	00	8E	32	C9	D7	I_>.IWēyI">.I2Ex
00000010	CB	7F	28	F4	86	22	3E	FF	8E	32	C9	FF	FF	FF	FF	FF	E.(6I">yI2Eyyyyy
00000020	F0	CB	E0	CC	C9	FF	FF	FF	C9	FF	FF	FF	FF	FF	FF	FF	8EaIEyyEyyyyyyy
00000030	C3	B2	31	FF	FF	FF	FF	FF	C3	70	31	FF	FF	FF	FF	FF	Å²1yyyyyÅp1yyyyy
00000040	C3	33	1B	FF	FF	FF	FF	FF	C3	1E	14	FF	FF	FF	FF	FF	Å3.yyyyyyÅ..yyyyy
00000050	C3	D9	1B	D9	FF	FF	FF	FF	FB	C3	60	1E	FF	FF	FF	FF	ÅU.UyyyyyÅ..yyyyy
00000060	D9	FF	FF	FF	FF	FF	FF	FF	C3	BD	2E	C3	B6	2E	C3	77	UyyyyyyÅy.Åy
00000070	2F	C3	A4	2A	C3	90	1F	C3	2D	31	C3	05	37	C3	1E	18	/Å*Å..Å-1Å.7Å..
00000080	C3	24	30	C3	1D	30	C3	9C	19	C3	B1	18	C3	35	14	C3	ÅS0Å.0ÅI.Å+.ÅS.Å
00000090	E2	2E	C3	00	13	C3	61	19	C3	77	19	01	01	02	03	04	Å.Å..Åa.Åw.....
000000A0	05	06	07	08	09	0A	0B	0C	0D	0E	0F	00	00	28	00	20	(.
000000B0	00	48	00	40	00	68	00	60	00	88	00	80	00	A8	00	A0	.H.@.h..I.e..
000000C0	00	C8	00	C0	00	E8	00	E0	00	FF	00	00	01	FF	00	20	.È.Å.e.å.y...y.
000000D0	01	FF	00	40	01	FF	00	60	01	FF	00	80	01	FF	00	A0	.y.@.y..y..I.y.
000000E0	01	FF	00	C0	01	FF	00	E0	01	FF	00	00	00	70	FF	40	.y.Å.y.å.y...py@
000000F0	FF	00	FF	C0	FE	80	FE	00	FE	80	FD	FF	FF	FF	FF	FF	y.yÅpI.pIyyyyyy
00000100	00	C3	50	01	CE	ED	66	66	CC	00	00	0B	03	73	00	83	ÅP.IiffI...s.I
00000110	00	0C	00	00	00	08	11	1F	88	89	00	0E	DC	CC	6E	E6	I.UIne
00000120	DD	DD	D9	99	BB	BB	67	63	6E	0E	EC	CC	DD	DC	99	9F	YŸU!gcn.iŸU!I
00000130	BB	B9	33	3E	53	4E	45	41	4B	59	20	53	4E	41	4B	45	»3»SNEAKY SNAKE
00000140	53	00	00	00	00	00	01	02	00	01	5D	00	0D	DD	BD	DD	S.....]ŸŸz
00000150	F3	AF	E0	80	C3	4E	15	97	0C	EC	11	5F	0E	34	12	D2	6aIÅN.I.i...4.0
00000160	0D	E9	0E	C6	0D	CE	0D	C3	0E	9B	0E	B5	07	2C	0C	FF	.é.Å.I.Å..I.µ...y
00000170	0D	93	0E	1D	0F	2D	0F	7B	0F	8F	0F	11	0F	9E	0D	A2	.I...- {...I.c
00000180	0D	A6	0D	AA	0D	AE	0D	B2	0D	B6	0D	E4	07	F2	11	BF	.].å.0.².Å.å.å.å.å
00000190	09	A5	0E	0A	12	20	12	36	0C	50	0C	0D	5F	39	0E	4B	Ÿ... .6.P.I.9.K
000001A0	0E	55	0E	8E	0B	B3	11	71	0E	7F	0E	20	08	DB	3C	C6	.U.I.³.q... .U<Å
000001B0	0B	EC	0B	FC	0B	08	0C	97	0E	49	0F	34	11	4C	11	44	.i.ü...I.I.4.L.D
000001C0	12	1D	6D	F1	0F	0F	10	2B	10	C0	06	F6	F4	F6	04	F6	Ÿ Ÿ Ÿ + Å Ÿ Ÿ Ÿ Ÿ

Questa protezione si rese necessaria per evitare che altri sviluppatori producessero software non licenziato da Nintendo. Come nota a margine c'è da dire che anche SEGA successivamente tentò lo stesso approccio con il Genesis, ma la sua causa contro la Accolade non andò come previsto:

https://en.wikipedia.org/wiki/Sega_v._Accolade.

COME BYPASSARE LA PROTEZIONE

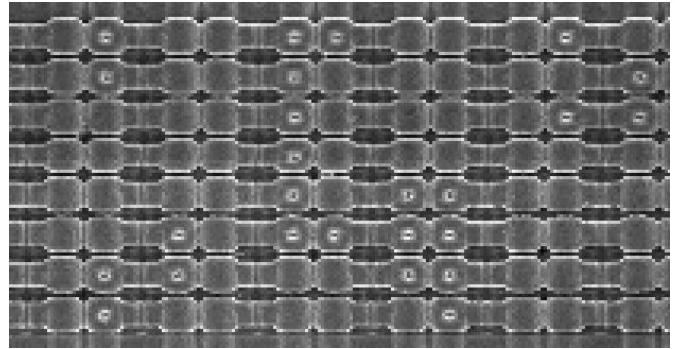
Il contenuto di questo mini-sistema operativo rimase un mistero fino al 2003 quando il coder Neviksti: (<http://dot-matrix-game.blogspot.se/2014/01/boot-roms.html>) effettuò un suo decapping (cioè una "scoperchiatura" fisica del chip) e lesse "ad occhio nudo" il suo contenuto, bit per bit (256x8 = 2048 bits totali), dopo averli fotografati tramite un microscopio; eccoli qui sotto in tutto il loro micro-splendore:



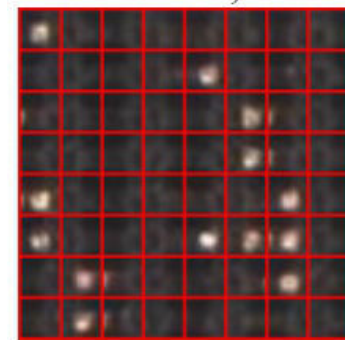
Ogni "pallino" rappresenta un bit a 1, l'assenza del pallino rappresenta un bit a 0. I pallini non sono altro che le cariche elettriche memorizzate nelle celle di memoria.

Eccovene un esempio ancora più zoomato ottenuto dal decap di un altro chip visto al microscopio elettronico

(nulla a che fare con le console):



Ecco un esempio di come vengono lette manualmente le cariche elettriche:



10000000
00001000
00000100
00000100
10000010
10001110
01000010
01000000

Davvero incredibile!

Una volta convertiti in byte e successivamente in ASM, il codice (commentato) della BootROM appare essere il seguente:

```
LD SP,$ffff ; $0000 Setup Stack
```

```
XOR A ; $0003 Zero the memory from $8000-$9FFF (VRAM)
```

```
LD HL,$9fff ; $0004
```

```
Addr_0007:
```

```
LD (HL-),A ; $0007
```

```
BIT 7,H ; $0008
```

```
JR NZ, Addr_0007 ; $000a
```

```
LD HL,$ff26 ; $000c Setup Audio
```

```
LD C,$11 ; $000f
```

```
LD A,$80 ; $0011
```

```
LD (HL-),A ; $0013
```

```
LD ($FF00+C),A ; $0014
```

```
INC C ; $0015
```

```
LD A,$f3 ; $0016
```

```
LD ($FF00+C),A ; $0018
```

```
LD (HL-),A ; $0019
```

```
LD A,$77 ; $001a
```

```
LD (HL),A ; $001c
```

```
LD A,$fc ; $001d Setup BG palette
```

```
LD ($FF00+$47),A ; $001f
```





```

LD DE,$0104 ; $0021 Convert and load logo
data from cart into Video RAM
LD HL,$8010 ; $0024
Addr_0027:
LD A,(DE) ; $0027
CALL $0095 ; $0028
CALL $0096 ; $002b
INC DE ; $002e
LD A,E ; $002f
CP $34 ; $0030
JR NZ, Addr_0027 ; $0032

LD DE,$00d8 ; $0034 Load 8 additional bytes
into Video RAM
LD B,$08 ; $0037
Addr_0039:
LD A,(DE) ; $0039
INC DE ; $003a
LD (HL+),A ; $003b
INC HL ; $003c
DEC B ; $003d
JR NZ, Addr_0039 ; $003e

LD A,$19 ; $0040 Setup background tilemap
LD ($9910),A ; $0042
LD HL,$992f ; $0045
Addr_0048:
LD C,$0c ; $0048
Addr_004A:
DEC A ; $004a
JR Z, Addr_0055 ; $004b
LD (HL-),A ; $004d
DEC C ; $004e
JR NZ, Addr_004A ; $004f
LD L,$0f ; $0051
JR Addr_0048 ; $0053

; === Scroll logo on screen, and play logo
sound===

Addr_0055:
LD H,A ; $0055 Initialize scroll count, H=0
LD A,$64 ; $0056
LD D,A ; $0058 set loop count, D=$64
LD ($FF00+$42),A ; $0059 Set vertical scroll
register
LD A,$91 ; $005b
LD ($FF00+$40),A ; $005d Turn on LCD, showing

Background
INC B ; $005f Set B=1
Addr_0060:
LD E,$02 ; $0060
Addr_0062:
LD C,$0c ; $0062
Addr_0064:
LD A,($FF00+$44) ; $0064 wait for screen frame
CP $90 ; $0066
JR NZ, Addr_0064 ; $0068
DEC C ; $006a
JR NZ, Addr_0064 ; $006b
DEC E ; $006d
JR NZ, Addr_0062 ; $006e

LD C,$13 ; $0070
INC H ; $0072 increment scroll count
LD A,H ; $0073
LD E,$83 ; $0074
CP $62 ; $0076 $62 counts in, play sound #1
JR Z, Addr_0080 ; $0078
LD E,$c1 ; $007a
CP $64 ; $007c
JR NZ, Addr_0086 ; $007e $64 counts in, play
sound #2
Addr_0080:
LD A,E ; $0080 play sound
LD ($FF00+C),A ; $0081
INC C ; $0082
LD A,$87 ; $0083
LD ($FF00+C),A ; $0085
Addr_0086:
LD A,($FF00+$42) ; $0086
SUB B ; $0088
LD ($FF00+$42),A ; $0089 scroll logo up if B=1
DEC D ; $008b
JR NZ, Addr_0060 ; $008c

DEC B ; $008e set B=0 first time
JR NZ, Addr_00E0 ; $008f ... next time,
cause jump to "Nintendo Logo check"

LD D,$20 ; $0091 use scrolling loop to pause
JR Addr_0060 ; $0093

; ===== Graphic routine =====
LD C,A ; $0095 "Double up" all the bits of

```





the graphics data

LD B,\$04 ; \$0096 and store in Video RAM

Addr_0098:

PUSH BC ; \$0098

RL C ; \$0099

RLA ; \$009b

POP BC ; \$009c

RL C ; \$009d

RLA ; \$009f

DEC B ; \$00a0

JR NZ, Addr_0098 ; \$00a1

LD (HL+),A ; \$00a3

INC HL ; \$00a4

LD (HL+),A ; \$00a5

INC HL ; \$00a6

RET ; \$00a7

Addr_00A8:

;Nintendo Logo

.DB \$CE,\$ED,\$66,\$66,\$CC,\$0D,\$00,\$0B,
\$03,\$73,\$00,\$83,\$00,\$0C,\$00,\$0D

.DB \$00,\$08,\$11,\$1F,\$88,\$89,\$00,\$0E,\$DC,\$CC,
\$6E,\$E6,\$DD,\$DD,\$D9,\$99

.DB \$BB,\$BB,\$67,\$63,\$6E,\$0E,\$EC,\$CC,\$DD,\$DC,
\$99,\$9F,\$BB,\$B9,\$33,\$3E

Addr_00D8:

;More video data

.DB \$3C,\$42,\$B9,\$A5,\$B9,\$A5,\$42,\$3C

; ===== Nintendo logo comparison routine =====

Addr_00E0:

LD HL,\$0104 ; \$00e0 ; point HL to Nintendo
logo in cart

LD DE,\$00a8 ; \$00e3 ; point DE to Nintendo
logo in DMG rom

Addr_00E6:

LD A,(DE) ; \$00e6

INC DE ; \$00e7

CP (HL) ; \$00e8 ;compare logo data in cart
to DMG rom

JR NZ,\$fe ; \$00e9 ;if not a match, lock up here

INC HL ; \$00eb

LD A,L ; \$00ec

CP \$34 ; \$00ed ;do this for \$30 bytes

JR NZ, Addr_00E6 ; \$00ef

LD B,\$19 ; \$00f1

LD A,B ; \$00f3

Addr_00F4:

ADD (HL) ; \$00f4

INC HL ; \$00f5

DEC B ; \$00f6

JR NZ, Addr_00F4 ; \$00f7

ADD (HL) ; \$00f9

JR NZ,\$fe ; \$00fa ; if \$19 + bytes from
\$0134-\$014D don't add to \$00

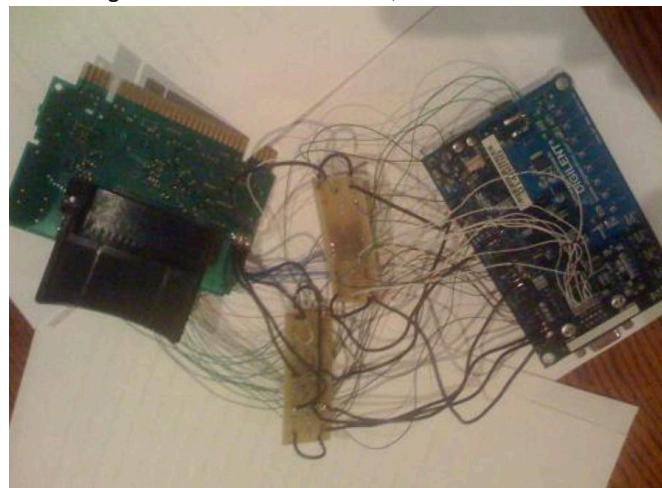
; ... lock up

LD A,\$01 ; \$00fc

LD (\$FF00+\$50),A ; \$00fe ;turn off DMG rom

Nel 2009 un altro reverser, **Costis Sideris**: https://www.its.caltech.edu/~costis/sgb_hack/, riuscì a dumpare la BootROM del Super Game Boy utilizzando, tramite un FPGA, un glitch hardware che disabilitava l'ultima istruzione di lock out della memoria dove era salvata la copia della BootROM, grazie all'invio di impulsi overclockati, riuscendo così a dumpare il codice dall'offset 0x0000 all'offset 0x00FF con dello specifico codice inserito dall'offset 0x0100 in poi. Ecco una foto del "semplice" sistema utilizzato:

Esaminando il codice di questa BootROM si è scoperto che, se il secondo checksum della sommatoria che deve essere uguale a 0 è diverso da 0, la cartuccia-adattatore



non disabilita la cartuccia di gioco, ma lo fa il Super Nintendo.

Sempre Costis Sideris e sempre nel 2009 riuscì a dumpare anche la BootROM del Game Boy Color, utilizzando una combinazione di glitch questa volta sia di clock che di alimentazione, attraverso un FPGA. In questo dump si è





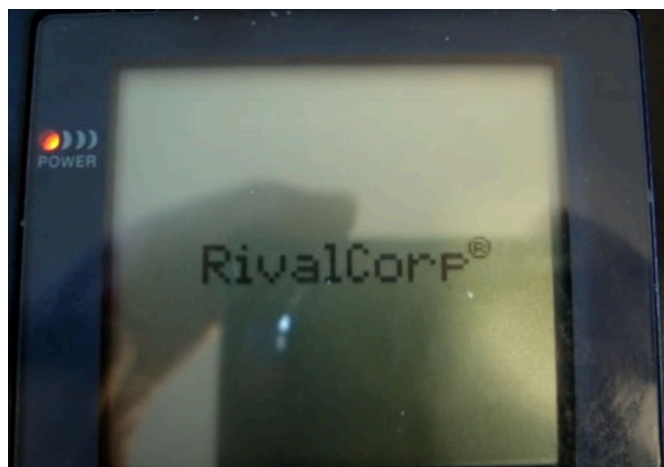
accorto che il check del logo avveniva solamente per la prima metà dei bytes e non di tutti e 48. Assieme ad essa riuscì a dumpare il contenuto di una seconda ROM di 1792 bytes presente in questa console.

Dr. Decapitator (alias **Christopher Tarnovsky**: https://en.wikipedia.org/wiki/Christopher_Tarnovsky) tentò un parziale decapping, identificando addirittura 3 ROMs (256, 1792 e 512 bytes) ma di quella da 512 non è stato possibile ottenere alcun dump (è possibile che contenga il microcodice della CPU oppure codice legato al LCD).

Si tratta dunque di segreti violati dopo più di un decennio. Inoltre nel 2012 è scaduto anche il brevetto (<https://www.google.com/patents/US5134391>) che descrive le funzioni della BootROM a confermare quanto scoperto precedentemente dai reverse engineer.

Una volta scoperto il funzionamento della BootROM alcuni sviluppatori si sono ingegnati per far sì che il Game Boy leggesse un logo custom dalla cartuccia al solito offset (prima fase che permette la visualizzazione del logo), ma poi verificasse un secondo logo (quello ufficiale), memorizzato ad un altro offset della cartuccia.

I risultati sono simili al seguente:



Il prossimo articolo ci porterà a scoprire le protezioni del **GameBoy Advance!**

Link utili

Loghi custom

A questo indirizzo potete trovare una bella descrizione delle varie tipologie di loghi custom prodotti da alcuni sviluppatori:
<http://fuji.drillspirits.net/?post=87>

BootROM

Ulteriori informazioni sulle BootROM, compresa quella trovata in una cartuccia prototipo di Pokemon Stadium per Nintendo 64, mancante nella release ufficiale della stessa:
http://gbdev.gg8.se/wiki/articles/Gameboy_Bootstrap_ROM

BootROM del GBC

Altre informazioni sulla BootROM del GBC le potete trovare:
https://tcrf.net/Game_Boy_Color_Bootstrap_ROM

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.





ZX99 e altri add-on non ufficiali per Sinclair ZX81

di Alberto Apostolo

Negli anni '80 del XX secolo, diverse aziende britanniche offrivano schede di espansione per aumentare la versatilità del modello ZX81.

La maggior parte delle schede incrementavano la RAM oppure erano circuiti aggiuntivi per effetti sonori o interfacce Joystick [SU83]. Altre consentivano di collegare stampanti professionali, salvare dati e programmi su floppy disk.

Nell'articolo sono descritte solo una parte di tutte le apparecchiature vendute all'epoca.

ZX99

ZX99 (Fig.1) era il nome di un sistema di controllo prodotto nel 1982 dalla Data-Asstette di Londra in vendita a 62.90 Sterline (VAT inclusa) e poi a 49.95 (Fig.2).

Il sistema poteva collegare fino a quattro registratori a nastro e in più aveva una interfaccia RS232 per collegare una stampante [Ada82].

Le routine per gestire i registratori erano memorizzate in una ROM speciale allocata nella sezione da 8K e 16K perché il sistema ZX99 considerava i registratori come unità di INPUT oppure unità di OUTPUT (due registratori erano il minimo consentito). Altri due cavi collegavano la scheda alle uscite EAR e MIC del Sinclair ZX81 (Fig.3). Per le operazioni di I/O occorreva lanciare le routine in linguaggio macchina tramite il comando USR del BASIC Sinclair. L'esito di tali operazioni si otteneva con il comando LET L = USR 1234 (L = zero indicava che tutto era andato bene). Sempre attraverso comandi BASIC e il lancio di routine l.m., si potevano salvare anche sequenze di byte. L'interfaccia RS232 poteva funzionare tra 110 e 9600 Baud



Fig.1

(Baud = bit per secondo).

Le indicazioni per scaricare il manuale sono in [DA99].

Infine, Data-Asstette proponeva altri dispositivi [DA98]:

- 1) ZX21 Metered Loading Interface a 19.95 per misurare la qualità del segnale,
- 2) ZX22 Jigsaw Tape Load Interface a 39.95 Sterline (con led di segnalazione del caricamento dei programmi),
- 3) ZX98 Parallel and serial interface a 99.95 Sterline per trasformare lo ZX81 in video-terminale e usufruire di servizi telematici.

MEMOPAK

Era scheda contenente una espansione di memoria prodotta della Memotech di Oxford (Fig. 4). Una specie di manuale (con programmi BASIC dimostrativi) si trova in [MEM93].

FORTH PER ZX81

La Skywave di David Husband (Bournemouth) produceva nel 1983 una espansione RAM e una ROM contenente il linguaggio FORTH (Fig.5). Il riferimento per scaricare il manuale è riportato in [FOR20].

FLOPPYDISC INTERFACE FOR ZX81

Il F.I.Z. (Floppydisc Interface for the ZX81, Fig.6,7,8) prodotto nel



Fig.2 ("ZX Computings" Oct-Nov. 1982, p.2, www.yumpu.com).



Fig.3

1982 dalla Macronics (Solihull, West Midlands) forniva nella stessa unità un floppy disk drive da 5.25 pollici, il software di gestione in 2K ROM [FIZ21][FIZ82].

Ogni disco aveva 43K di spazio con una velocità di caricamento di 8K in 22 secondi ed era suddiviso in 34 tracce da 1280 bytes ($34 \times 1280 = 43520$).

Le routine di gestione contenute nella ROM di 2K si potevano richiamare dai programmi BASIC. L'unità forniva anche la corrente per l'intero sistema (compreso il Sinclair ZX81). Era prevista anche la vendita di altre interfacce (di cui una per creare una rete locale) e una stampante.

F.I.Z. era venduto a 303.03 Sterline (VAT inclusa). La sola motherboard costava 79.95 Sterline (più 1.50 Sterline per le spese di spedizione) ed era compatibile con l'espansione Memopak 64 [DA98] ma non ci sono conferme sul fatto che funzionasse veramente con quella espansione.





MEMOTECH Explores the Excellence of your ZX81

THIS MONTH'S
2 NEW PACKS!

CENTRONICS I/F



Memotech's Memopak Range

All five of the currently available Memopaks are housed in elegant black anodised aluminium cases, and are styled to fit wobble-free into the back of the ZX81, allowing more add-ons (from Memotech or Sinclair) to be connected.

- MEMOPAK 64K MEMORY EXTENSION**
The 64K Memopak extends the memory of the ZX81 by 16K, and with the ZX81 gives 64K, which is neither switched nor paged and is directly addressable. The unit is user transparent and accepts commands such as 10 DIM A(5000). Breakdown of memory areas: 0-8K Sinclair ROM; 8-16K This area can be used to hold machine code for communication between programmes or peripherals; 16-64K-A straight 48K for normal Basic use.
- MEMOPAK 32K and 16K MEMORY EXTENSIONS**
These two packs extend and complete the Memotech RAM range (for the time being!) A notable feature of the 32K pack is that it will run in tandem with the Sinclair 16K memory extension to give 48K RAM total.
- MEMOPAK HIGH RES GRAPHICS PACK**
H/RG Main Features — • Fully programmable Hi-Res (192 x 248 pixels) • Video page is both memory and bit mapped and can be located anywhere in RAM • Number of Video pages is limited only by RAM size (each takes about 6.5K RAM) • Instant inverse video on/off gives flashing characters • Video pages can be superimposed • Video page access is similar to Basic plot/print commands • Contains 2K EPROM monitor with full range of graphics subroutines controlled by machine code or user function
- MEMOPAK CENTRONICS TYPE PARALLEL PRINTER INTERFACE**
Main Features — • Interfaces ZX81 and parallel printers of the Centronics type • Enables use of a range of dot matrix and daisy wheel printers with ZX81 • Compatible with ZX81 Basic, prints from LIST, LPRINT and COPY • Contains firmware to convert ZX81 characters to ASCII code • Gives lower-case characters from ZX81 inverse character set

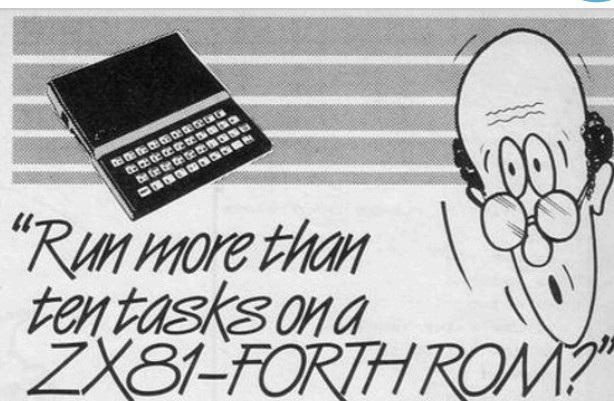
Coming Soon...



We want to be sure you are satisfied with your Memopak — so we offer a 14-day money back Guarantee on all our products. Memotech Limited, 3 Collins Street, Oxford OX4 1XL, England Tel: Oxford (0865) 722102 Telex: 837220 Orchid G

ZX COMPUTING OCT/NOV 1982

Fig.4 ("ZX Computings" Oct-Nov. 1982, p.11, www.yumpu.com).



"Run more than ten tasks on a ZX81-FORTH ROM?"

Sure! More than 10 tasks simultaneously and, in some cases, up to 300 times faster! That's what replacing the basic ROM with the new FORTH does for the ZX81 — and more!

The brains behind the breakthrough belong to David Husband, and he's building Skywave Software on the strength of it. Already orders are flooding in and it's easy to see why.

The ZX81-FORTH ROM gives you a totally new system. In addition to multi-tasking and split screen window capability, you can also edit a program while three or four others are executing, schedule tasks to run from 50 times a second to once a year, and with a further modification switch between FORTH and BASIC whenever you like.

The ZX81-FORTH ROM gives you a normal keyboard with a 64 character buffer and repeat, it supports the 16k, 32k, 64k RAM packs, it is fig-FORTH compatible and it supports the ZX printer.

The price, too, is almost unbelievable. As a "fit it yourself Eprom", complete with manual, it's just £25+VAT.

Add £2 p&p UK (£5 Europe, £10 outside Europe) and send your order to the address below.

Skywave SOFTWARE

David Husband
73 Curzon Road, Bournemouth,
BH1 4PW, ENGLAND.
Tel: (0202) 302385.
International +44 202 302385.

Fig.5 [SP84]

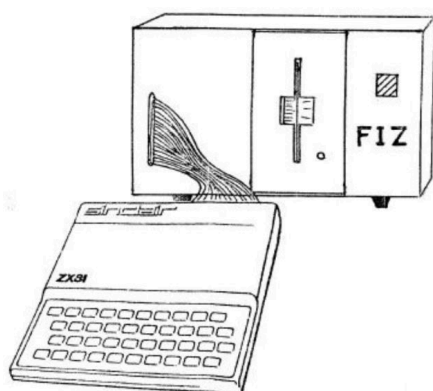


Fig.6

MEMORY MAP

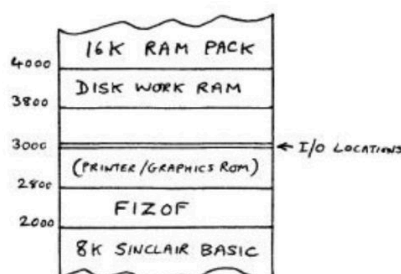


Fig.7

PHYSICAL LAYOUT

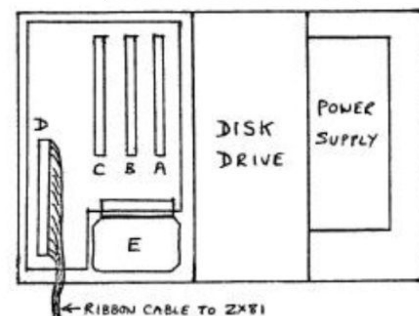


Fig.8

Bibliografia

- [Ada82] S.Adams, "Control system boosts power", Sinclair User n.5, Aug. 1982, <https://www.sinclairuser.com/005/zx99.htm>
- [DA98] Data-assette (comm.adv.), Sync n.3, May-Jun 1983, pag.116, https://archive.org/details/syncmagazine-v3_n3
- [DA99] Data-assette, "ZX99 User Manual 2K Rom", Interact Software Limited, 1982, <https://archive.org/details/zx99-user-manual>
- [FIZ21] FIZ, retrieved (2021/08/14) from <https://spectrumcomputing.co.uk/entry/1000151/Hardware/FIZ>
- [FIZ82] Sinclair User n.2, May 1982, (2020/08/19) retrieved from <https://www.sinclairuser.com/002/hardware.htm>
- [FOR20] Skywave Software, "Sinclair ZX81-FORTH Rom with Multi-Tasking" (2010/08/19) retrieved from <http://zx81.ordi5.free.fr/dominique/download/zx81-forth-h4th.pdf>
- [MEM93] Fanzine ZXirQLiveAlive n.4, 1993, pagg.7-8, https://archive.org/details/ZXirQLiveAlive_Issue_0304/mode/2up
- [SP84] Sinclair Programs n.18, Apr 1984, <https://datassette.nyc3.cdn.digitaloceanspaces.com/revistas/sinclairprograms18-apr84.pdf>
- [SU83] Sinclair User n.12, Mar 1983, http://www.retro8bitcomputers.co.uk/Content/downloads/magazines/sinclairuser/SinclairUser_012_Mar_1983.pdf





Analogue Pocket il palmare multi-videogioco rivoluzionario!

di Franco Bressan

Analogue è una compagnia che si è fatta un nome all'interno della comunità retro non solo per l'alto prezzo dei suoi dispositivi ma anche per la perfetta riproduzione dei giochi attraverso FPGA che tentano di riprodurre il funzionamento delle console originale senza sfruttare gli emulatori.

L'Analogue Pocket mantiene questo concetto, avere l'esatto stesso funzionamento delle console portatili di Nintendo e, attraverso moduli venduti separatamente, Game Gear, Neo Geo Pocket, Atari Lynx e TurboGrafx-16 senza mai emulare niente.

Rimanendo puramente nel tentare di volere giocare ai giochi Game Boy Color/Advance, perché dovrei scegliere l'Analogue pocket invece del portatile Nintendo?

Per l'elemento principale del Pocket: lo schermo.

Monta uno schermo LCD da 3 polici e mezzo ad una risoluzione di 1600x1440, esattamente 10 volte la risoluzione dell'originale a 8 bit.

Questo permette di scalare perfettamente i nostri titoli preferiti in alta risoluzione, con una fedele riproduzione

dei colori e il tutto in uno schermo retroilluminato di alta qualità. Altri comfort che il Pocket introduce è la possibilità di mettere il gioco standby e riprendere direttamente da dove ci eravamo fermati, in futuri aggiornamenti verrà inserita anche la possibilità di creare delle save state, di ricreare le caratteristiche del display del hardware originale e per chi si diletta nello sviluppo di giochi Game Boy, Analogue permette di elaborare nuovi titoli con GB Studio che ha aggiunto il Pocket tra i dispositivi compatibili, ha anche un DAW integrato per la produzione delle colonne sonore.

Il pocket funziona anche con adattatori per cartucce di altri sistemi come Game Gear, Neo Geo Pocket, Atari Lynx e molte altre. Un progetto che non emula ma è in FPGA.

Tra le caratteristiche che spiccano di più troviamo anche la possibilità di fare musica. Il pocket possiede una workstation audio digitale chiamata Nanoloop. E' un sintetizzatore e un sequencer progettato per la creazione di musica e per performance dal vivo. Una gioia per gli





amanti della chiptune.

Inoltre è possibile collegarlo al PC, Mac o altro attraverso i cavi Midi che troverete in vendita a parte.

Per finire è possibile collegare la console al televisore con un adattatore HDMI. Potete collegare in wireless i controlli 8BitDo Bluetooth oppure i controlli tramite usb cablato.

SPECIFICHE TECNICHE:

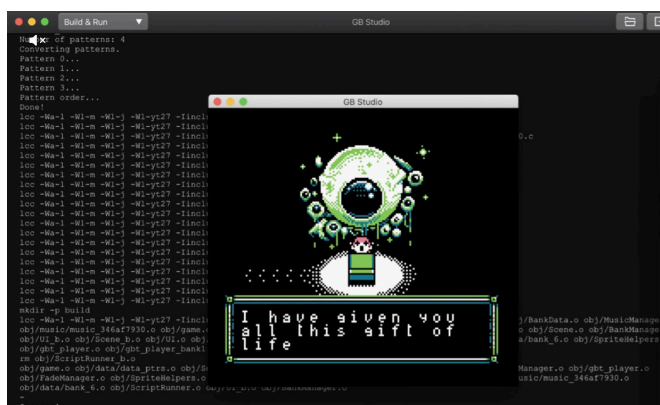
Console

- Compatibile con le cartucce di gioco per Game Boy, Game Boy Color e Game Boy Advance
- LCD da 3,5". Risoluzione 1600×1440. 615ppi.
- Rotazione del display a 360° (modalità tate)
- Batteria ricaricabile agli ioni di litio da 4300 mAh
- 6-10 ore di gioco e oltre 10 ore di sonno
- Tutti i pulsanti mappabili
- Altoparlanti stereo
- slot per scheda micro SD
- Ricarica USB-C
- Porta di collegamento in stile originale
- Uscita cuffie da 3,5 mm



Dock

- Uscita HDMI 1080p
- Bluetooth e supporto 2.4g per controller wireless
- 2 ingressi USB per controller cablati
- Supporto fino a 4 giocatori (4p Bluetooth, 2p 2.4g, 2p USB cablato)
- Compatibile DAC
- Power Dock acceso dal controller



Per maggiori informazioni e ordini potete consultare il sito web ufficiale: <https://www.analogue.co/pocket>





Fairchild Channel F

di Leonardo Miliani

Nello scorso articolo abbiamo esaminato la genesi dei dispositivi da gioco domestici andando a sviscerare tutti i segreti di quella che è in assoluto la prima console domestica messa in commercio, la Odyssey di Magnavox. Seguendo quel filo conduttore, in questo articolo andremo a fare la conoscenza di una piccola star, a modo suo. La protagonista della nostra storia vanta diversi primati: è reputata la prima console della seconda generazione, è la prima console ad utilizzare un microprocessore, ed è la prima console basata su cartucce con memorie ROM intercambiabili per caricare i giochi. Stiamo parlando della Fairchild Channel F (fig. 1).

RAVEN

Per capire bene le origini della Channel F dobbiamo partire dal contesto in cui nacque l'idea che portò alla sua realizzazione e per fare ciò dobbiamo tornare indietro di diversi decenni, dobbiamo tornare negli USA degli anni '60. In questo periodo a Stamford, Connecticut, c'è il dipartimento di ricerca e sviluppo della AMF (American Machine & Foundry), una società dalle mille sfaccettature, che produce di tutto, dagli arredi per giardino ai reattori nucleari (!), ed in questa sede vengono prodotti gli schermi segna-punteggio per le sale da bowling. Ad un certo punto agli impiegati che lavorano lì arriva la notizia del trasferimento della sede nella Carolina del Nord. Non tutti sono entusiasti di spostarsi di 500 miglia in un altro Stato ed alcuni decidono di lasciare l'azienda. Tra questi c'è un ingegnere di nome Norman Alpert che, insieme ad altri ex-dipendenti AMF, fonda la Alpex Computer. Inizialmente

la società si dedica alla produzione di registratori di cassa elettronici ma gli affari lentamente prendono una brutta piega e la Alpex Computer, agli inizi degli anni '70, si ritrova in piena crisi. Un altro ex-dipendente di AMF, Wallace Kirschner, matura l'idea che una possibile strada da percorrere per evitare la bancarotta sia quella appena aperta nell'emergente mercato dei videogiochi elettronici. Questa idea prende corpo vedendo come la prima console domestica per videogiochi, la Magnavox Odyssey, stia riscuotendo un buon successo, e di come l'arcade Pong di Atari stia letteralmente spopolando nei bar e nei pub.

Questi giochi hanno però un difetto: non si possono cambiare. Il Pong, e gli arcade che lo seguono a ruota, hanno una scheda madre composta da decine di circuiti logici che creano il videogioco sullo schermo e gestiscono l'interazione con il giocatore ma se il produttore vuole cambiare gioco questi è costretto a riprogettare da zero la scheda. Anche la Odyssey, per quanto "programmabile" tramite schede esterne, è in realtà una console con componenti passivi e le schede gioco funzionano alla stregua di switch ed interruttori per cambiare alcune linee elettriche e variare così la logica di funzionamento della console, proponendo varianti dell'unico schema di gioco integrato nella stessa. Ma le cose stanno per cambiare, perché in quegli anni Intel presenta nel 1971 il 4004, il primo microprocessore monolitico della storia, seguito l'anno successivo dall'8008. Si tratta di una piccola rivoluzione perché finalmente si riesce a concentrare un sacco di transistor in un unico contenitore di piccole dimensioni creando un componente che può caricare ed eseguire istruzioni lette da una memoria esterna il cui contenuto può essere modificato per cambiare il programma eseguito. Kirschner capisce che questa sia la strada giusta e propone la sua idea ad Alpert il quale avalla immediatamente il progetto di un videogioco ma capisce anche che si tratta di una corsa a tempo contro altri concorrenti che si stanno gettando in quel settore e che questa corsa non può vincerla con i pochi impiegati della sua azienda. Decide quindi di assumere Lawrence Haskel, un ingegnere con cui ha già lavorato in passato ai tempi di AMF. Haskel ha anche un vantaggio, è un appassionato



Fig. 1: La Fairchild Video Entertainment System, poi divenuta Channel F (foto di Evan Amos - fonte: Wikimedia Commons)



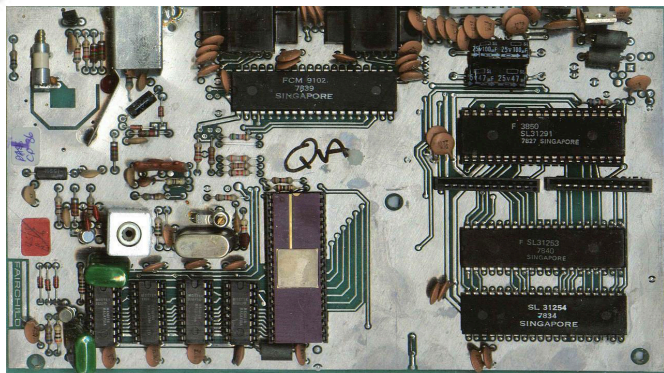


Fig. 2 - La scheda madre della Channel F System II. Da notare i pochi circuiti integrati presenti (foto: Incog88 - fonte: Wikimedia Commons)

egli stesso di videogiochi per cui per lui è un sogno che si avvera il poter lavorare ad un tale progetto. È così che agli inizi del 1974 Kirschner ed Haskel iniziano a lavorare ad un progetto che chiamano “Remote Access Video Entertainment”, o RAVEN: il primo si dedica alla parte hardware mentre il secondo programma i giochi. Come “cervello elettronico” scelgono proprio l’8008, il primo processore ad 8 bit mai prodotto, a cui affiancano un circuito in grado di generare un’immagine di 128x64 pixel su un comune schermo TV. Una così bassa risoluzione è dettata dal fatto che all’epoca le memorie sono veramente care e per gestire un’immagine di 8.192 pixel totali ($128 \times 64 = 8192$) servono 8 kilobit di memoria (8.192 bit = 1 Kbyte) dove ogni bit rappresenta un pixel (acceso o spento) sullo schermo. Durante lo sviluppo, Kirschner adotta il più potente processore 8080 non appena questo viene messo in commercio da Intel.

Hasckel, da parte sua, realizza il primo gioco ispirandosi al Pong di Atari ma facendolo più ricco e più simile all’hockey, uno sport molto praticato nei freddi inverni del nord-est americano. Realizza quindi un campo di gioco tratteggiandolo con bordi e linee di area, aggiunge delle racchette che non solo si possono muovere in verticale ma anche in orizzontale, potendo quindi essere spostate per tutta l’area di gioco. Inoltre aggiunge la possibilità di ruotarle così da far variare il rimbalzo della pallina a seconda della posizione della racchetta. Aggiunge anche dei “parastinchi del portiere” (se avete mai visto una partita di hockey, avete bene a mente quelle protezioni legate sugli stinchi dei portieri), delle barriere che possono essere spostate solo in verticale per difendere la propria porta. Dopo l’ “Hockey” programma il “Tic-Tac-Toe” (il tris), lo “Shooting Gallery” (dove il giocatore deve ruotare una “pistola” e sparare ad un bersaglio mobile) e “Doodle”, un primitivo programma di arte dove il giocatore deve

tracciare o cancellare delle righe sullo schermo. Per permettere tutti i movimenti degli elementi dei suoi giochi, Kirschner realizza due gamepad con numerosi pulsanti, ad ognuno dei quali corrisponde un comando tipo andare in alto, in basso, ruotare, sparare, ecc.

Per risolvere la questione dei giochi intercambiabili viene adottata una soluzione molto semplice. Acquistano da RadioShack dei contenitori plastici rettangolari con delle dimensioni molto simili a quelle delle comuni musicassette dell’epoca ed al loro interno alloggiano una piccola scheda con su montata una EPROM: ogni cartuccia ne contiene una diversa, per ognuno dei giochi realizzati, e si inserisce in un connettore della scheda madre della console. La scelta delle EPROM viene loro dalla scheda di sviluppo che Intel distribuisce per l’8080, scheda contenente appunto questo tipo di memoria che può essere cancellata e riprogrammata per correggere errori del programma o per caricare nuove versioni dello stesso.

Da RAVEN a Fairchild Video Entertainment System

Il prototipo viene presentato ad Alpert che lo trova molto interessante ma capisce anche che la sua piccola azienda non avrebbe mai potuto trovare le risorse per poterla commercializzare autonomamente. Decide quindi di trovare un partner commerciale, e per primi vengono contattati alcuni produttori di apparecchi TV, pensando che un prodotto del genere da vendere abbinato ai loro televisori potrebbe destare il loro interesse. Ma le cose non vanno come sperato perché, nonostante diverse dimostrazioni, nessuno vuole investirci sopra. Alpert decide quindi di contattare Fairchild Semiconductor, un grosso produttore di integrati da cui acquista regolarmente componenti per i suoi prodotti. Viene mandato un ingegnere a visionare il RAVEN, che viene reputato interessante: la relazione incuriosisce anche il vice-presidente del reparto consumer il quale a sua volta parla con l’amministratore

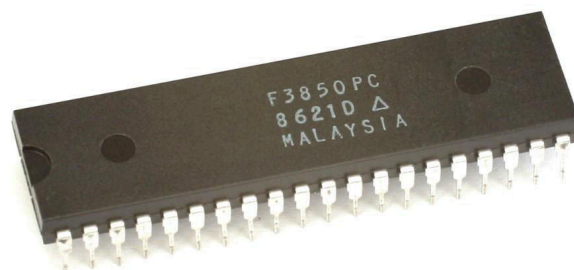


Fig. 3 - La CPU Fairchild F8 (foto: Konstantin Lanzet - fonte: Wikimedia Commons)



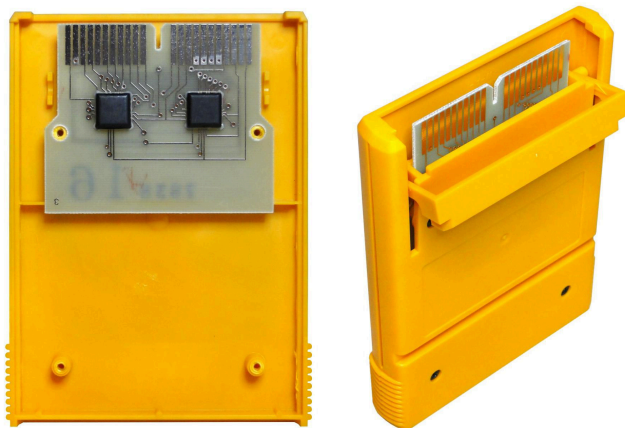


Fig. 4 - L'interno di una Videocart e lo sportellino di protezione dei contatti (foto: Schnurrikowski- fonte: Wikimedia Commons)

delegato Wilf Corrigan di Fairchild Camera and Instrument, la casa madre. Corrigan spedisce alla Alpex Computer Gene Landrum e Jerry Lawson per valutare a fondo il prototipo, che viene valutato commercialmente interessante. Fairchild stringe quindi una collaborazione con Alpert per finalizzare la console e trasformare il RAVEN in una console da videogiochi vera e propria, affiancando per questa fase a Kirschner e Haskel proprio Lawson. Prima di tutto viene adottata la CPU F8 di Fairchild (fig. 3), requisito essenziale per poter instaurare la collaborazione (d'altronde, un produttore di microprocessori non avrebbe mai messo in commercio un prodotto basato su una CPU di una diretta concorrente...), e successivamente Lawson lavora per semplificare il sistema di controllo basato sulla tastiera usata sul RAVEN affidando il progetto a Nicholas Talesfore che realizza l'iconico joystick verticale con testina mobile. Il progetto intanto viene ribattezzato "STRATOS".

Dopo queste modifiche, STRATOS viene approvato per la commercializzazione e si inizia a progettare la versione definitiva che deve avere dimensioni più ridotte rispetto al RAVEN, tali da permettere di poter essere appoggiato sopra ad un comune TV. Un altro punto da sistemare sono le cartucce. Accantonato il contenitore rabberciato usato da Kirschner e Haskel, Talesfore affida il compito a Ron Smith, un ingegnere meccanico con cui ha lavorato in passato e che ha già realizzato un sistema di memorie intercambiabili per una calcolatrice di National Semiconductor, anche se era stata cancellata dall'azienda prima di arrivare in commercio. Le cartucce vengono progettate per essere resistenti agli urti e con uno sportellino che copre il pettine con i pin di collegamento che si apre automaticamente una volta inserita nell'apposito

alloggiamento nella console: una volta nella loro sede, un meccanismo di blocco le tiene ancorate per evitare che vengano sfilate inavvertitamente. Per renderle familiari agli acquirenti della console, vengono realizzate in modo che abbiano circa le dimensioni delle cassette Stereo8 allora in uso per la musica e realizzate di un appariscente colore giallo in modo che siano facilmente identificabili (fig. 4). Il contenitore della console viene realizzato in legno e plexiglass scuro con i pulsanti per l'accensione, il reset e la selezione dei giochi integrati nonché le varie modalità sul pannello anteriore. Particolare degno di nota è la presenza di un pulsante di pause per fermare temporaneamente il gioco in esecuzione. Sulla parte posteriore del lato superiore vengono ricavati 2 alloggiamenti per riporre i joystick quando non in uso: questi ultimi non sono separabili dalla console ma fissati in modo permanente con cavi non scollegabili.

La console nella sua veste finale fa il primo debutto ufficiale al Summer Consumer Electronics Show di Chicago nel mese di giugno del 1976 come Video Entertainment System, o VES. Durante l'evento non attira però l'attenzione di stampa e visitatori perché ciò che Fairchild mostra è un semplice guscio vuoto, senza elettronica funzionante. Dopo quella prima apparizione fa una comparsa più dignitosa in un articolo di Businessweek del 6 luglio 1976 intitolato "The Smart Machine Revolution" dove vengono mostrate le futuristiche ed innumerevoli applicazioni degli emergenti microprocessori, apparendo accanto ad altri dispositivi elettronici. Il vero debutto commerciale avviene verso la fine dell'anno: la console viene pubblicizzata su carta stampata e in televisione e messa in vendita al prezzo di 169,95 dollari. Le vendite vanno inizialmente bene perché il salto tecnico rispetto alla precedente Odyssey è notevole, e la possibilità di avere nuovi giochi oltre a quelli che si possono trovare preinstallati attira gli acquirenti.

Analisi tecnica

La console è costruita intorno al microprocessore Fairchild F8. Questo, sviluppato nel 1971, si tratta in realtà di un "chipset" ossia di un insieme di integrati che affiancati vanno a formare il processore vero e proprio. La configurazione utilizzata nella prima versione della console vede l'impiego del Fairchild 3850 e del Fairchild 3851 (fig. 2). Il primo integrato è la CPU vera e propria, contenente l'ALU (l'unità aritmetica logica), 2 porte di input/output e 64 byte di RAM disponibile sotto forma di



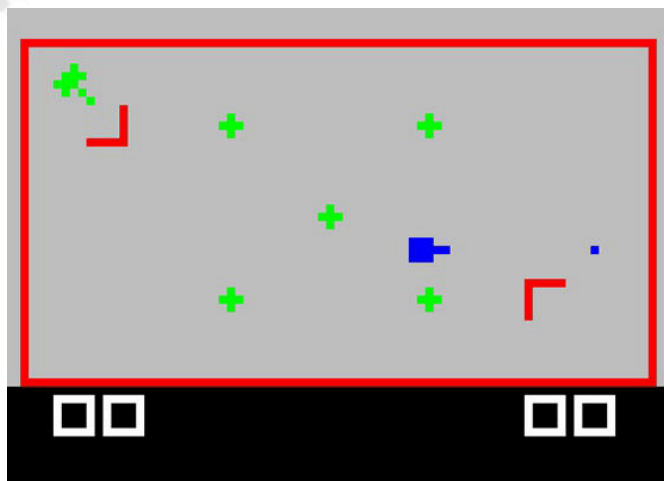


Fig. 5 - Il gioco Desert Fox (fonte: voxodyyssey.com)

registri interni: oltre a questi byte la CPU non ha altra RAM dedicata, per cui sono i giochi stessi a fornire memoria extra quando necessario, come ad esempio "Maze" e "Chess". L'integrato 3851 è invece denominato PSU, per "Program Storage Unit", e contiene 1 KB di ROM, altre 2 porte di input/output, un bus di controllo, la gestione di un interrupt, il registro di stack, il Program Counter che punta in memoria la successiva istruzione da eseguire. Nella ROM sono memorizzati non solo il BIOS per gestire la console all'accensione e avviare i giochi sulla cartuccia ma anche dei giochi integrati. Siccome la filosofia che aveva portato alla realizzazione dell'F8 era quella di ridurre le caratteristiche di ogni chip allo stretto necessario, la configurazione minima utilizzabile era proprio quella proposta sulla console, composta dalla CPU 3850 e dal PSU 3851. In questi 2 integrati c'è tutto quanto serve per realizzare un microcomputer: CPU, ALU, RAM, ROM e Program Counter. E' il PSU che recupera dalla sua ROM interna l'istruzione da eseguire e la passa alla CPU tramite un bus dedicato: per questo motivo né il 3850 né il 3851 sono dotati di bus indirizzi esterno. Il PSU possiede però un "bus di comando" a 5 linee specifico per gestire particolari ROM progettate per lavorare con l'F8 e dotate di un proprio Program Counter interno che gestiscono esse stesse. Il PSU perciò spedisce "comandi" per recuperare i dati dalla ROM e queste ultime recuperano i dati richiesti, a seconda del comando ricevuto e del valore del loro Program Counter. Le prime cartucce giochi sono dotate di 2 chip di queste speciali ROM da 1 KB l'uno, collegate alla console mediante un connettore a 22 pin dove passano le linee necessarie allo scambio dei dati. La frequenza della CPU è 1,79 MHz, ottenuta dimezzando il clock del generatore del segnale video del colore (color burst), di 3,58 MHz.

Rispetto al RAVEN, il buffer video viene portato a 2 KB, grazie a cui la console può generare un'immagine video con una risoluzione massima di 128x64 pixel e 2 bit colore per ogni pixel (4 colori da una tavolozza di 8): salire troppo con la risoluzione avrebbe comportato il dover usare più memoria con un considerevole aumento del costo di produzione. La VRAM (video RAM) è composta da 4 chip DRAM 4096 che offrono ognuno 4 Kbit di capacità, per un totale di 16 Kbit bit, ossia 2 Kbyte ($128 \times 64 = 8.192$ pixel = 8.192×2 bit = 16.384 bit = 2.048 byte). Non tutti i pixel sono però utilizzabili né visualizzabili. Le celle 125 e 126 controllano infatti la tavolozza in uso sulla corrispondente riga dello schermo mentre l'immagine vera e propria parte dalla riga 4 e colonna 4 e si estende per 102 pixel in orizzontale e 58 pixel in verticale: queste sono quindi le dimensioni effettive dell'immagine sullo schermo. A questo si aggiunge che non tutti i televisori dell'epoca riuscivano ad aggiustare la larghezza dell'immagine per visualizzare l'immagine per cui spesso le porzioni più esterne in orizzontale andavano perse, riducendo ulteriormente la dimensione dello schermo a 95x58 pixel. L'immagine visualizza pixel bianchi se si imposta lo sfondo su nero oppure verdi, rossi e blu se si sceglie uno sfondo colorato.

I joystick sono un vero e proprio prodigio elettromeccanico. La testina triangolare non solo legge le 8 direzioni orizzontali classiche ma può anche essere ruotata in senso orario o antiorario, come nei paddle in stile Pong per capirsi, ed inoltre può essere sia tirata verso l'alto che premuta verso il basso, offrendo quindi tutta una serie di comandi mai offerta da nessun altro controller giochi fino ad allora visto in commercio. L'audio è limitato a 3 toni con 3

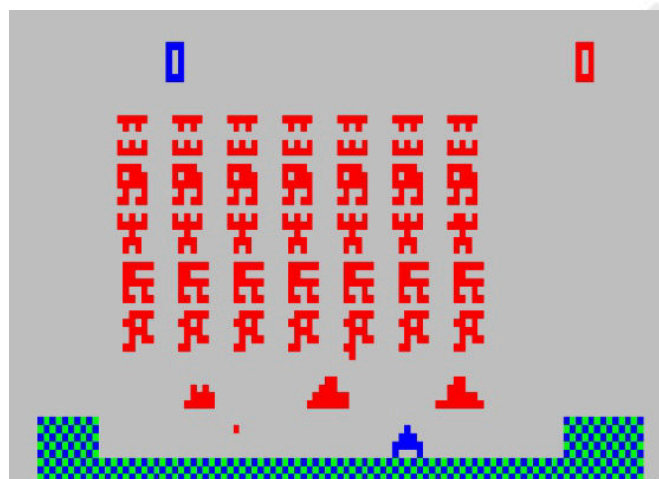


Fig. 6 - Alien Invasion, un ovvio clone di Space Invaders (font: voxodyyssey.com)



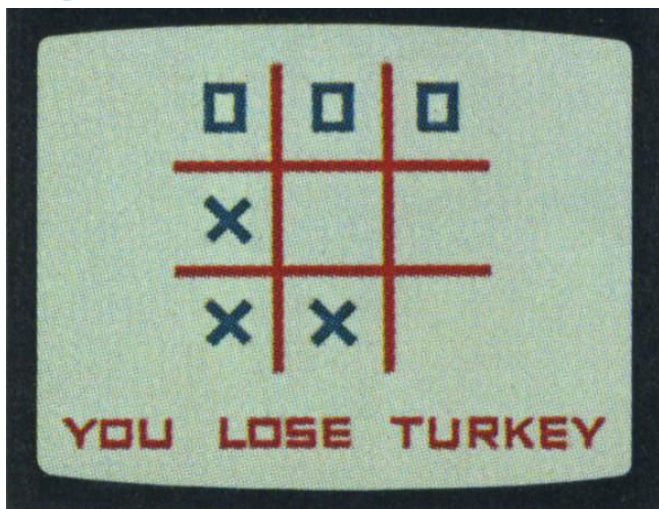


Fig. 7 - "Hai perso, tacchino!"
(fonte FastCompany.com)

frequenze fisse a 120 Hz, 500 Hz e 1 KHz, riprodotti da un altoparlante interno.

I giochi disponibili inizialmente sono cinque, ossia quelli inizialmente sviluppati da Hasckel: 2 sono contenuti nella ROM della console stessa, "Tennis" e "Hockey", e 3 sono pubblicati su una cartuccia chiamata "Videocart-1": "Tic-Tac-Toe", "Shooting Gallery" e "Doodle". E proprio durante la conversione del codice dall'assembly 8080 a quello per F8 Lawrence ha inserito in "Tic-Tac-Toe" quello che è reputato essere il primo scherzo contenuto all'interno di un videogioco. In quel periodo, come detto, Lawson è alla guida del progetto di conversione del RAVEN in STRATOS ed uno dei suoi epiteti preferiti con cui appella le persone è "turkey", tacchino. Lawrence decide di riprendere quell'epiteto ed inserirlo nel gioco in modo che appaia sullo schermo nel caso in cui il giocatore perda la partita (fig. 7). Curiosamente, Lawson non si è accorto subito della cosa ma molto tempo dopo: pare comunque che l'abbia presa bene, per cui lo scherzo ha funzionato! A questi giochi seguono, nel corso degli anni in cui la console è rimasta in produzione, altri titoli pubblicati sempre in esclusiva da Fairchild prima e da Zircon, che rileverà i diritti sulla console, successivamente, per un totale di 27 cartucce (alcune con più di un gioco). Inizialmente ogni cartuccia è messa in vendita al prezzo di 19,95 dollari l'una, un prezzo non indifferente se si rapporta al costo iniziale della console.

Channel F e derivati

L'uscita della VES spinse diversi produttori ad accelerare lo sviluppo dei propri sistemi per evitare di accumulare eccessivo ritardo nei confronti della console di Fairchild.

Tra le concorrenti va segnalata l'Atari Video Computer System (successivamente divenuta 2600) la cui uscita spinse Fairchild a cambiare nome al proprio sistema per evitare confusione negli acquirenti, divenendo "Channel F", dove la "F" era l'abbreviazione di "Fun" (divertimento, in inglese). L'uscita dell'Atari VCS segnò anche l'inizio del declino della Channel F, surclassata dalla rivale in termini di vendite. La VCS di Atari veleggiava a ritmi inimmaginabili: se già nel primo anno di commercializzazione aveva toccato le 500.000 unità, nel solo 1979 ne furono vendute un milione di unità. Per contro, fino al 1979 la Channel F aveva venduto complessivamente 350.000 unità... Fairchild lentamente perse interesse nei confronti del mercato dei videogiochi e nel 1979 cedette i diritti della console ad una società denominata Zircon International. La lotta con la VCS era impari. Per rendere la console più accattivante e cercare di recuperare in parte il divario, Zircon rivide l'aspetto esterno della Channel F realizzando un design più compatto e moderno e dotando la console di controller separabili dal corpo macchina. L'audio fu migliorato, eliminando l'altoparlante interno e miscelandolo con il segnale RF per farlo riprodurre attraverso la TV. La nuova unità prese il nome di "Channel F System II". Fu anche data in licenza a diversi produttori per ampliare la penetrazione su altri mercati, soprattutto quello europeo (fig. 8). Ciò però non bastò a recuperare quote di mercato e la console fu tolta definitivamente dal commercio nel 1983, quando ormai i suoi giochi venivano giudicati come "antiquati" e la grafica ed il sonoro "primitivi".

L'insuccesso commerciale

Il problema di fondo della Channel F è stato l'essere realizzata da una azienda che non aveva come primo obiettivo quello di realizzare giochi. La Channel F è stata



Fig. 8 - La Luxor Video Entertainment Computer, versione per il mercato svedese (foto: liftarn - fonte: Wikimedia Commons)





sviluppata da Fairchild Camera and Instrument, una multinazionale che spaziava per ogni dove con diverse aziende controllate: la gestione della console é stata affidata alla Exetron, la divisione responsabile dei prodotti consumer, che acquistava le CPU F8 dalla Fairchild Semiconductor, la divisione dedicata alla produzione dei componenti elettronici e dei circuiti integrati. La casa madre vedeva infatti la console come principalmente un veicolo per far vendere chip alla Fairchild Semiconductor. Atari, invece, era nata come società votata esclusivamente alla produzione di videogiochi, ed ogni suo sforzo era deputato alla realizzazione del miglior gioco possibile. Ed infatti i titoli prodotti per la 2600, nonostante le limitazioni hardware della macchina, surclassavano di diversi ordini di grandezza quelli della concorrenza, Channel F compresa: la 2600 offriva titoli dinamici, adrenalinici, pieni di colori e effetti sonori nonché ricchi di azione mentre la Channel F offriva giochi “classici”, lenti, monotoni, troppo “calmi” per gli affamati videogiocatori dell’epoca che stavano iniziando ad esplorare un mondo appena scoperto. Il parco giochi della 2600 cresceva mensilmente, mentre quello della Channel F vide la produzione di qualche decina di titoli in 7 anni di vita: non ci poteva essere confronto. E non ci fu.

Eredità

Come detto in apertura, la Channel F vanta diversi primati fra cui quello di essere stata la prima console basata su un microprocessore, nello specifico il Fairchild F8. L’avere a bordo un microprocessore ha permesso alla console di poter essere “programmata”, ossia di poter eseguire programmi anche complessi: la Odyssey sposta le racchette sullo schermo semplicemente seguendo i comandi del giocatore ma qui c’è un “cervello elettronico” che permette di eseguire dei calcoli in autonomia e dotare la console di una primitiva forma di intelligenza artificiale sufficiente ad offrire, per la prima volta, la possibilità di giocare “contro il computer” e non più solamente contro un altro giocatore umano. “Tic-Tac-Toe”, “Baseball” e “Checkers” sono esempi in cui l’avversario da battere è mosso dalla CPU.

Anche le cartucce sono state usate per la prima volta in ambito video-ludico proprio dalla Channel F. In verità le cartucce intercambiabili non erano una novità perché prima di allora erano già state usate per permettere di cambiare il software di alcune calcolatrici realizzate da diversi produttori, tra cui National Semiconductor, Texas Instruments, HP ed altri, ma era la prima volta che un tale sistema veniva adottato per un accessorio che sarebbe stato manipolato da migliaia, decine di migliaia di persone senza particolari accortezze e per chissà quante volte al giorno. Le cartucce furono una vera e propria incognita, una scommessa dei progettisti: nessuno aveva dati riguardo a quanto sarebbero potute “sopravvivere” in mano a ragazzi e bambini, non c’erano dati statistici su cui basarsi per fare previsioni. Per questo furono studiate e progettate per essere le più robuste possibili, e dotate di ogni espediente per ridurre i possibili danneggiamenti come il sistema di blocco quando inserite o lo sportellino a proteggere il pettine dei contatti che si apriva automaticamente al loro inserimento. Ma da quella idea le cartucce si sono diffuse così tanto da risultare per più di due decenni il sistema di distribuzione esclusivo dei giochi di tutte le console venute dopo la Channel F, dettando legge fino a metà degli anni ‘90 quando i CD prima ed i DVD poi hanno iniziato a sostituirle come media più economico e capiente. E non sono mai sparite del tutto perché le console portatili continuano ad usarle anche ai giorni nostri.

Insomma, oggi la Channel F è una signora attempata che però in gioventù ha fatto girare molto la testa.





Il linguaggio FOCAL sul PDP-8 - parte 1

di Francesco Fiorentini

Nel numero scorso, mostrandovi come riprodurre il gioco Planet Raith in ambiente GEOS con il BeckerBASIC, vi ho raccontato la storia del suo antenato, King of Sumeria, un gioco scritto nel 1968 su PDP-8 utilizzando il linguaggio FOCAL. Purtroppo in quell'articolo non ho potuto approfondire l'argomento e vorrei quindi rimediare con queste pagine.

Il PDP-8

Il PDP-8 è un minicomputer con architettura a 12bit prodotto dalla **DEC, Digital Equipment Corporation** negli anni sessanta e, come si evince dal nome, l'ottavo progetto della serie DEC **PDP, Programmed Data Processor**.

La serie PDP deve il suo nome ad una curiosa decisione di marketing, strettamente correlata al nome dell'azienda. **Ken Olson** e **Harlan Anderson** fondarono la DEC nel 1957 ed inizialmente volevano che si chiamasse Digital Computer Corporation con l'intenzione di costruire computer. Gli altri azionisti però insistettero per togliere la parola

computer dal nome della società e per attendere nella costruzione degli stessi. Questo perché, in un momento in cui i computer erano enormi, costosi e necessitavano di personale altamente specializzato, non credevano ancora in questo business. Per la stessa ragione, quando i primi modelli della serie PDP-1 furono prodotti e messi in vendita, ad un prezzo di poco superiore ai 100.000 dollari mentre i computer dell'epoca costavano ancora milioni di dollari, fu evitato di utilizzare la parola computer.

La serie PDP ottenne degli ottimi risultati ed il PDP-8 riscosse un grande successo di vendite, fu infatti acquistato da scuole, università e centri di ricerca, soprattutto negli Stati Uniti, facendo registrare quasi **300.000** unità vendute. Grazie a questo risultato il PDP-8 è stato paragonato alla Ford T, essendo stato il primo computer ad essere prodotto in massa ed ad un costo accessibile a (quasi) tutti.

Tecnicamente il PDP-8 utilizza uno spazio di indirizzamento della memoria a 12 bit, così la configurazione di base ha una memoria principale di 4.096 parole a dodici bit. Le istruzioni hanno un opcode di 3 bit e ci sono quindi solo

Opcode			Indirect	Page	Address							
11	10	09	08	07	06	05	04	03	02	01	00	

otto istruzioni.

Ci sono solo tre registri visibili al programmatore: un accumulatore a 12 bit (AC), un contatore di programma (PC) e un flag di riporto chiamato "link register" (L).

Le prime versioni del PDP-8 non erano corredate di sistema operativo ed i programmi venivano inseriti direttamente in codice macchina binario tramite gli interruttori presenti sul pannello frontale.

Con il passare degli anni furono sviluppati sistemi operativi come OS/8 e COS-310 corredate da un editor in linea, che permisero lo sviluppo di compilatori a riga di comando per linguaggi come l'assembly **PAL-III**, il **FORTRAN**, il **BASIC**, il **DIBOL** (Digital's Business Oriented Language) ed il **FOCAL**.



Fig. 1 - Model DEC PDP-8 minicomputer, foto da Science Museum Group Collection

Il linguaggio FOCAL



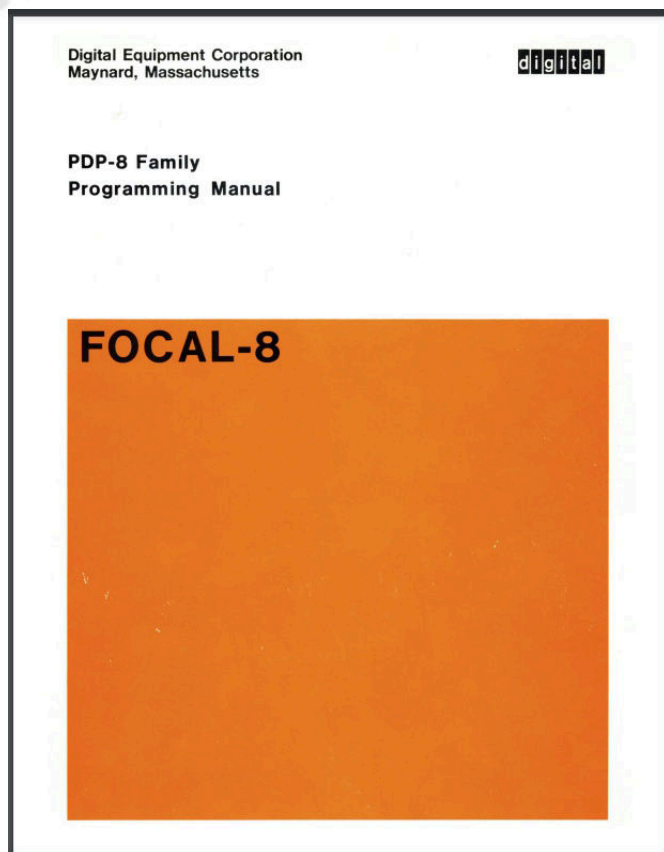


Fig. 2 - Il manuale di programmazione del FOCAL-8

FOCAL, acronimo di **FOR**mula **CAL**culator, è un linguaggio di programmazione interpretato, sviluppato da DEC per la sua serie di macchine PDP. Il linguaggio fa la sua prima apparizione nel 1968 sui minicomputer PDP-8 ed assume il nome di **FOCAL-8**. Un anno più tardi viene rilasciata una versione aggiornata dell'interprete chiamata **FOCAL-69**. Questa è la versione che adopereremo nel nostro tutorial più avanti.

Il linguaggio FOCAL è stato evidentemente influenzato da un altro linguaggio, il **JOSS**, acronimo di **JOHNNIAC Open Shop System** che era stato implementato con successo sulle macchine della serie PDP-6. Come il JOSS e successivamente il BASIC, di cui il FOCAL è uno degli antenati, FOCAL sul PDP-8 è un ambiente di programmazione completo, comprendente editor, interprete e routine di I/O.

Il linguaggio è composto da comandi imperativi ed espressioni matematiche generalmente in notazione standard, risultando quindi molto facile da apprendere e maneggiare. Come nel BASIC, è possibile utilizzare alcuni comandi anche in modalità immediata, senza specificare quindi il numero di linea e ricevendo immediatamente il risultato una volta premuto enter.

Una particolarità del linguaggio FOCAL, o almeno del FOCAL-8 nella sua configurazione originale, è che non gestisce le stringhe ed accetta valori alfanumerici in input, per esempio per assegnare i valori alle variabili, soltanto per trasformarli in numeri. Vedremo qualche esempio di questo comportamento più avanti.

Un po' di pratica...

Dopo avervi presentato il PDP-8 ed il linguaggio FOCAL, come mio consueto però voglio mostrarvi praticamente come mettere le mani su tale linguaggio e scrivere qualche semplice programma, come se fossimo alla fine degli anni sessanta, ben prima che la maggior parte delle macchine di cui scriviamo di solito fossero state costruite.

Grazie agli emulatori oggi giorno possiamo virtualmente mettere le mani su quasi tutti i computer costruiti dall'uomo e, ovviamente, il PDP-8 non fa eccezioni.

Tra i tanti emulatori presenti ho scelto questo realizzato in python per un paio di motivi. Prima di tutto sembra funzionare bene e secondariamente il sorgente è in chiaro, cosa che eventualmente permetterebbe a chi ne avesse voglia, di metterci le mani sopra per produrne un fork oppure per apportare piccole modifiche come ho fatto io. Personalmente ho trovato l'immagine del pannello frontale eccessivamente ingombrante e quindi ho deciso di dimezzarne la dimensione, modificando il codice dell'emulatore per gestire la nuova impostazione (se qualcuno fosse interessato a questa modifica, mi scriva pure).

Occorrente ed istruzioni

Python:

- <https://www.python.org/downloads/>

Libreria pygame:

- <https://www.pygame.org/news>

Emulatore e programmi PDP-8:

- <https://github.com/mdoege/PDP-8/>

Una volta scaricato ed installato python, lanciate questo comando da riga di comando per installare automaticamente la libreria pygame senza bisogno di scaricarla precedentemente:

```
python3 -m pip install -U pygame --user
```

Scaricate l'emulatore PDP-8 scritto in python da mdoege dal suo repository github: <https://github.com/mdoege/PDP-8/>





Posizionatevi all'interno della cartella dove avete salvato i file scaricati da github e, per lanciare l'emulatore di PDP-8, lanciate il seguente comando:

```
python pdp8.py
```

A questo punto per caricare l'interprete FOCAL lanciate i seguenti comandi:

```
PC 0000 AC 0000 L 0 SW 0000 IE 0
> load focal    per caricare il modulo FOCAL
PC 0200 AC 0000 L 0 SW 0000 IE 0
> r            per eseguire il modulo FOCAL
```

Se tutto é configurato correttamente dovreste trovarvi di fronte a qualcosa di simile a Fig. 3.

Il PDP-8 vi informará che l'interprete **FOCAL 1969** é stato lanciato con successo sul vostro emulatore PDP-8:

CONGRATULATIONS!!

YOU HAVE SUCCESSFULLY LOADED 'FOCAL,1969' ON A PDP-8 COMPUTER.

facendovi inoltre 2 domande:

```
SHALL I RETAIN LOG, EXP, ATN ?:NO
SHALL I RETAIN SINE, COSINE ?:NO
PROCEED.
*
```

L'interprete vi mostra le opzioni disponibili all'utente per il mantenimento delle funzioni matematiche. Se queste funzioni non saranno necessarie per il programma che avete intenzioni di realizzare, potete disattivarle rispondendo NO e FOCAL cancellará queste funzioni dal nucleo; così

da liberare lo spazio disponibile per i programmi utente.

Oggi come oggi, questa opzione fa sorridere, ma su un PDP-8, dove lo spazio era decisamente tiranno, era un'opzione necessaria.

Ma quanto spazio guadagnavamo esattamente eliminando queste funzioni dalla memoria? Possiamo accertarcene utilizzando il comando LOCATIONS:

*LOCATIONS

3206

3217

3217

5377

Hhmmm, ok, adesso probabilmente ne sappiamo ancora meno di prima... Cosa significano questi numeri?

Per fortuna il manuale **FOCAL-8 PROGRAMMING MANUAL FOR PDP-8/1** viene in nostro aiuto. L'esecuzione del comando LOCATIONS fa sì che FOCAL stampi quattro numeri ottali (la memoria del core del PDP-8 funziona su numeri in base 8) che rappresentano le seguenti posizioni all'interno del nucleo:

- a. **3206** -> inizio del buffer di testo
- b. **3217** -> fine del buffer di testo
- c. **3217** -> fine della lista delle variabili
- d. **5377** -> limite della memoria disponibile

Per calcolare lo spazio a disposizione dell'utente per i suoi programmi dobbiamo sottrarre c da d.

Quindi **d-c -> 5377 - 3217 = 2160**.

Attenzione però, 2160 é un numero in **base 8**, che tradotto

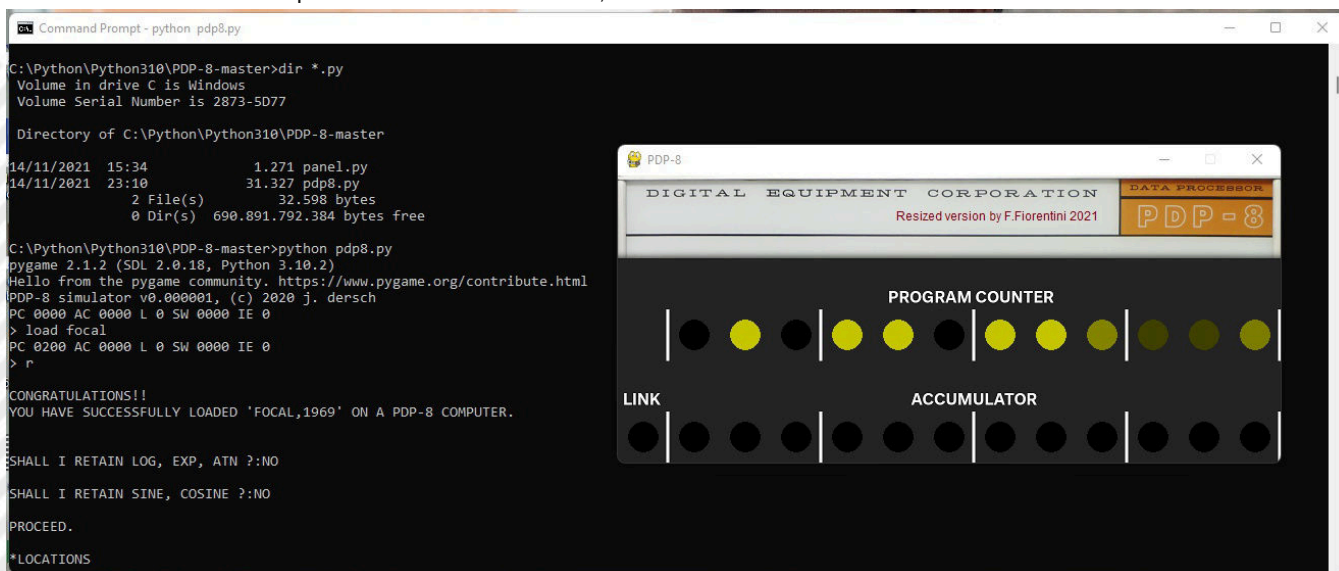


Fig. 3 - L'emulatore PDP-8 in funzione con il pannello in primo piano





in base 10 é uguale a 1136. Abbiamo quindi a disposizione circa 1100 caratteri per il nostro programma.

Questo però perché abbiamo scelto di eliminare le funzioni

Dialogue response	Yes/Yes	No/Yes	No/No
Locations	*L	*L	*L
	3206	3206	3206
	3217	3217	3217
	3217	3217	3217
	4617	5177	5377

matematiche, altrimenti il nostro spazio sarebbe stato molto più risicato, si veda la seguente tabella:

Mantenendo tutte le funzioni matematiche lo spazio a nostra disposizione si sarebbe ridotto a circa 700 caratteri, circa 900 in caso di mantenimento delle funzioni LOG, EXP, ATN e circa 1100 nel caso di nessuna funzione matematica mantenuta, come abbiamo visto precedentemente.

I comandi del FOCAL-8

La lista dei comandi di base a disposizione del programmatore FOCAL é decisamente limitata, niente a che vedere con le centinaia di comandi a disposizione dei linguaggi moderni e ben distante anche da quelli dei BASIC più poveri (vedi BASIC V2 del Commodore 64).

Ecco la lista dei comandi: **TYPE, ASK, WRITE, SET, ERASE, GO, GOTO, DO, IF, RETURN, QUIT, COMMENT, FOR, MODIFY.**

Il comando TYPE

Type é uno dei comandi più utilizzati in FOCAL. Permette infatti di calcolare e stampare a video il risultato di un'operazione o più semplicemente di una variabile e funziona anche come comando immediato. Proviamo a fare un esempio:

```
*TYPE 23/17
= 1.3530*
```

In questo caso abbiamo chiesto di dividere 23 per 17 ed il risultato mostrato é 1.3530. Di default infatti l'interprete FOCAL é programmato per stampare a video un valore numerico formato da 4 interi e 4 decimali (gli zeri non necessari vengono soppressi a favore di uno spazio).

Provate adesso a digitare la seguente istruzione:

```
*TYPE 23/17, 34+59
= 1.3530= 93.0000*
```

Come potete notare possiamo calcolare più espressioni in una sola riga, separandole con una virgola. Un'altra

cosa che avrete probabilmente già notato é che il comando TYPE non forza un carriage return dopo aver stampato il risultato. Per inserire un 'a capo' dopo ogni espressione ed alla fine del nostro calcolo, possiamo utilizzare il ! (punto esclamativo).

```
*TYPE 23/17,!,34+59,!
= 1.3530
= 93.0000
*
```

Il comando TYPE però ci permette anche di specificare il formato in cui visualizzare il risultato. Nel caso volessimo aumentare il numero di decimali possiamo utilizzare la seguente notazione:

```
TYPE %x.yz
```

Dove x é il numero di caratteri totali in output e yz é il numero di decimali

```
*TYPE %9.06,23/17,!,34+59,!
= 1.352940
= 93.000000
*TYPE %10.05,23/17,!
= 1.35294
```

Ovviamente la priorità viene sempre data ai numeri significativi.

```
*TYPE %9.06,1234+9999
= 11233.0000*
```

Bene, per il momento mi fermo qui, credo di avervi proposto già diversi spunti interessanti per cominciare ad approfondire questo linguaggio da soli, ovviamente se lo vorrete.

Se invece siete curiosi ma non avete tempo o voglia di studiare da soli, appuntamento al prossimo numero per continuare insieme la nostra avventura con il linguaggio

Link utili

Riporto qui due link ad altrettanti manuali FOCAL.

FOCAL-8 PROGRAMMING MANUAL FOR PDP-8/1, PDP-S/I, PDP-B/S, PDP-8, LINC-8, PDP-12, PDP-5:
http://bitsavers.informatik.uni-stuttgart.de/pdf/dec/pdp8/focal/DEC-08-AJAB-D_FOCAL_Programming_Manual_Jan70.pdf

FOCAL Programming Manual for PDP-S PDP-SIS PDP-S/I LAB-S LINC-S:
<http://www.bitsavers.org/www.computer.museum.uq.edu.au/pdf/DEC-08-AJAB-D%20PDP-8-I%20FOCAL%20Programming%20Manual.pdf>





Frattali in BASIC in salsa 8-bit - Parte 3

di David La Monaca

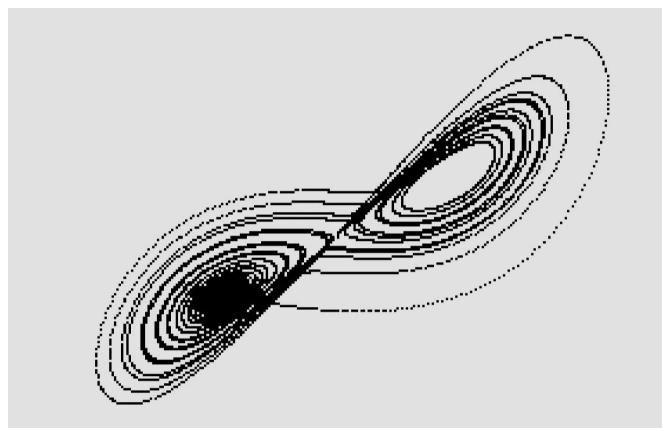
Siamo arrivati al terzo appuntamento di questa rubrica e lo stupore davanti a ciò che sono capaci di creare i nostri piccoli home computer a 8 bit con il loro BASIC e qualche algoritmo ben architettato resta immutato. Chiaramente il tallone d'Achille del nostro divertimento è la lentezza dell'elaborazione e del calcolo dei punti e delle linee da tracciare in alta risoluzione. Molti dei programmi che abbiamo pubblicato finora non sono ottimizzati per la massima velocità ed efficienza, soprattutto perché l'obiettivo di questo nostro piccolo viaggio è divertirsi nel vedere come queste macchine siano in grado di generare i classici frattali ed altre figure geometriche 2D o 3D di generale interesse. Per chi non possiede le macchine originali ed usa emulatori o sistemi FPGA, talvolta è disponibile una funzione "Warp Mode", molto utile per accelerare la velocità d'esecuzione dei nostri listati.

Ringraziamo tutti gli appassionati lettori di RMW che anche stavolta ci hanno spedito i risultati delle loro opere. In questo numero accogliamo altri algoritmi classici forniti da **Armando Pavese** per C64 e Vic-20 (ovviamente "dopati" con le espansioni Simons' Basic e SuperExpander Japan VIC-1211M) ed una particolare soluzione per Commodore Plus/4 e 128 per disegnare la Spirale di Ulam, basata sulla disposizione dei numeri primi, che ci è stata proposta da **Giuseppe Mignogna**.

Vale la pena di spendere qualche parola su questo particolare algoritmo e sul modo in cui è stato implementato da Giuseppe. La spirale di Ulam è un diagramma composto dalla sequenza dei numeri interi positivi disposti secondo un ordine preciso, per l'appunto a spirale. In questa particolare disposizione, la posizione dei numeri primi rivela una trama grafica molto interessante che ancora oggi è oggetto di studio. La trovata fu del matematico polacco Stanislaw Ulam che scarabocchiando i numeri interi secondo una spirale (vedi fig. A) si accorse che i numeri primi tendevano a disporsi lungo alcune diagonali del diagramma. Il fatto che i numeri primi si dispongano lungo le diagonali non è di per sé un fatto particolare perché ovviamente tutti i numeri primi tranne il 2 sono numeri dispari. Ciò che fu di interesse per Ulam e per una lunga serie di matematici che nel tempo si sono avvicinati al problema è che i numeri primi tendono a disporsi lungo *certe diagonali* e non altre. Alcuni studi rigorosi hanno stabilito che la concentrazione dei numeri primi è effettivamente maggiore su alcune diagonali. Dal punto

di vista visuale ed estetico, che è quello che ci interessa di più in questa rubrica, l'algoritmo che genera la spirale di Ulam produce un piacevole effetto grafico se si colorano i numeri primi e si lasciano per così dire "spenti" i non primi. L'effetto potete ammirarlo nelle versioni per Plus/4 e 128 ed il listato BASIC pubblicato è eseguibile su entrambe le macchine Commodore a 8-bit senza modifiche. Per ragioni di completezza e per onorare il suo lavoro, diciamo subito che Giuseppe ci ha inviato sia il listato che vedete in queste pagine ma anche una versione in cui si supera, alla maniera dei programmatori pratici di un tempo, il problema della lentezza dei calcolatori a 8-bit nel computare gli N numeri primi utili. Il listato qui presentato calcola effettivamente i numeri primi necessari alla colorazione, mentre nell'altra versione presenta un elenco precaricato di numeri primi fino al 32159 tramite linee DATA. Ovviamente il secondo listato, che non pubblichiamo per via della sua relativa lunghezza, disegna la spirale e la completa nello schermo ad alta risoluzione del Plus/4 e del C128 in un tempo di gran lunga inferiore. Per chi lo desidera il programma è a disposizione dei lettori su richiesta.

Ed ora, coraggio e mettiamo mano alle tastiere! Ecco di seguito un'ulteriore selezione di frattali e di figure geometriche per i vostri e i nostri amati 8-bit!



12. DRAGON 32 - Attrattore di Lorenz (piano XY)

```
10 PMODE 4,1:SCREEN 1,1
20 COLOR 0,1:PCLS
30 R=10:S=28:B=8/3:D=.002
40 X=.1:Y=0:Z=0
50 FOR I=1 TO 16000
60 PSET (127+5.7*X,95-3*Y)
70 XX=X+D*R*(Y-X)
80 YY=Y+D*(X*(S-Z)-Y)
```

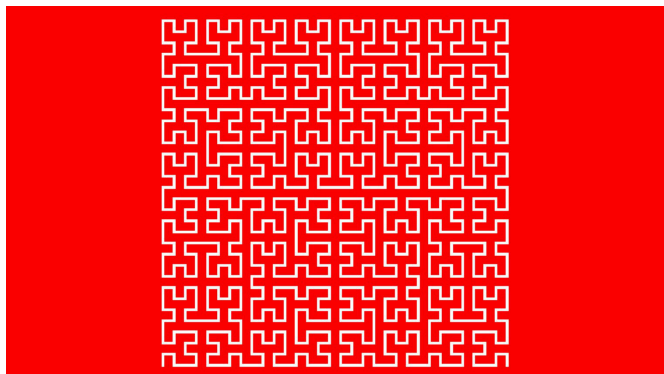




```

90 Z=Z+D*(X*Y-B*Z)
100 X=XX:Y=YY
110 NEXT
120 GO TO 120

```

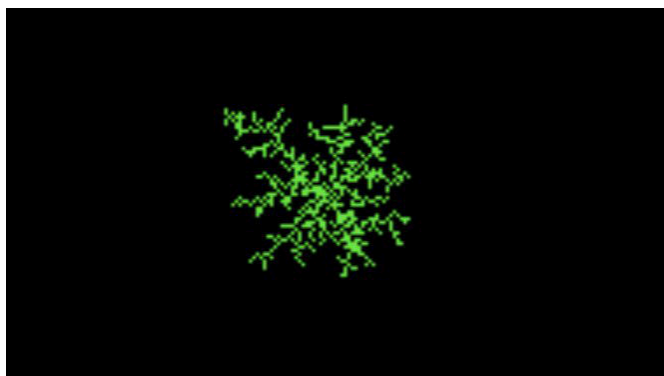


13. ORIC-1 - Curva di Hilbert

```

10 HIRES:PAPER 1
20 DIM S(50)
30 CURSET 57,165,1
40 S(0)=5:S(1)=1:SP=2:H=4:V=0
50 GOSUB 70
60 END
70 SP=SP-2:N=S(SP)-1:A=S(SP+1)
80 IF N=-1 THEN RETURN
90 S(SP)=A:S(SP+1)=N:S(SP+2)=-A
100 S(SP+3)=-A:S(SP+4)=N:S(SP+5)=A
110 S(SP+6)=N:S(SP+7)=A
120 S(SP+8)=-A:S(SP+9)=N:S(SP+10)=-A
130 S(SP+11)=A:SP=SP+12
140 GOSUB 170:GOSUB 70:DRAW H,V,1
145 GOSUB 170:GOSUB 70:DRAW H,V,1
150 GOSUB 70:GOSUB 170:DRAW H,V,1
155 GOSUB 70:GOSUB 170
160 RETURN
170 SP=SP-1:A=S(SP):T=-A*H:H=A*V:V=T
180 RETURN

```



14. ZX Spectrum Basic - Albero browniano

```

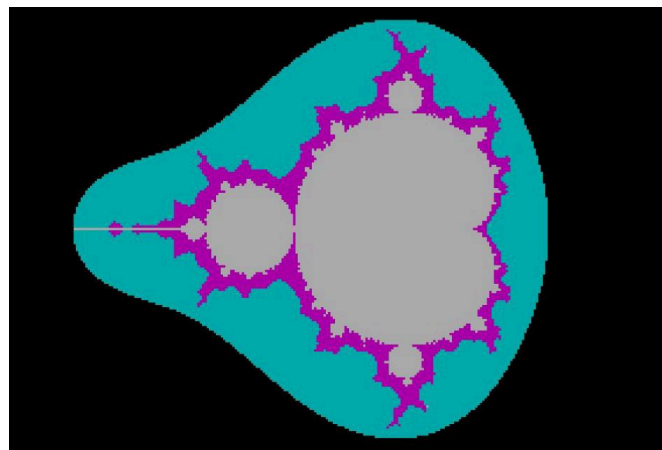
10 INK 4: PAPER 0: BORDER 0: BRIGHT 1: CLS
20 RANDOMIZE

```

```

30 PLOT 127,87
40 FOR i=8 TO 32 STEP 4
50 FOR j=1 TO 4*i
60 LET x=127+INT (2*i*RND) -i
70 LET y=87+INT (2*i*RND) -i
80 IF POINT (x,y) THEN GO TO 60
90 PLOT x,y
100 LET n=0
110 FOR a=x-1 TO x+1
120 FOR b=y-1 TO y+1
130 LET n=n+POINT (a,b)
140 NEXT b
150 NEXT a
160 IF n>1 THEN GO TO 250
170 PLOT INVERSE 1;x,y
180 LET x=x+INT (3*RND)-1
190 LET y=y+INT (3*RND)-1
200 IF x>=127+i THEN LET x=127-i+1: GO TO 220
210 IF x<=127-i THEN LET x=127+i-1
220 IF y>=87+i THEN LET y=87-i+1: GO TO 90
230 IF y<=87-i THEN LET y=87+i-1
240 GO TO 90
250 NEXT j
260 NEXT i

```



15. IBM PC 8088 BASIC, 1981 - Mandelbrot set

(Il listato può anche essere compattato in una singola linea!)

```

10 key off:screen 1
20 for i=-160 to 160
30 for j=-96 to 96
40 x=i/76:y=j/76
50 a=0:b=0:k=0
60 while a*a+b*b<4 and k<200
70 c=a*a-b*b+x:b=2*a*b+y:a=c:k=k+1
80 wend
90 pset(185+i,100+j),0-(k=200)-(k>=10)-(k>=4)
100 next
110 next

```





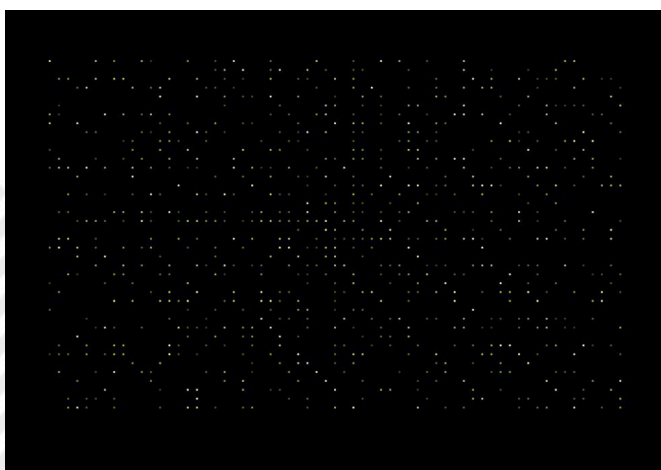
```
120 while -1:wend
```



16. VIC-20 + SUPER EXPANDER JP VIC-121 1M (6502, 1981) - La felce di Barnsley

(adattamento di Armando Pavese)

```
10 hires0:setc6,1,7
20 x=0:y=0
30 fori=-3500to3500
40 r=rnd(1)
50 ifr<.01thenx=0:y=.16*y:goto90
60 ifr<.86thent=.85*x+.04*y:y=-.04*x+.
85*y+1.6:x=t:goto90
70 ifr<.93thent=.2*x-.26*y:y=.23*x+.
22*y+1.6:x=t:goto90
80 t=-.15*x+.28*y:y=.26*x+.24*y+.44:x=t
90 plot(25*y)/1.45,90-25*x
100 nexti
110 wait198,1
120 text:end
```



17. C16-Plus/4 (prima immagine) & C128 (seconda immagine) - Spirale di Ulam

(di Giuseppe Mignogna - versione base)

```
10 color0,1:color4,1:graphic1,1
20 x=160:y=100:c=0:d=1:p=5:np=1
30 fori=1to2
40 forj=0toc
```

```
50 gosub200
55 color1,8,1:l=1+1:ifl=8thenl=0
60 ifr=1thendrawl,x,y
70 ifd=1thenx=x+p
80 ifd=2theny=y-p
90 ifd=3thenx=x-p
100 ifd=4theny=y+p
120 np=np+1
130 next
140 d=d+1:ifd=5thend=1
150 next
160 c=c+1
170 goto30
180 rem sub verifica se np = nro primo
190 rem risultato in r (0=false,1=true)
200 ifnp=1thenr=1:goto300
210 r=0:dv=3
220 ifnp/dv=int(np/dv) thenr=0:goto300
230 dv=dv+2
```

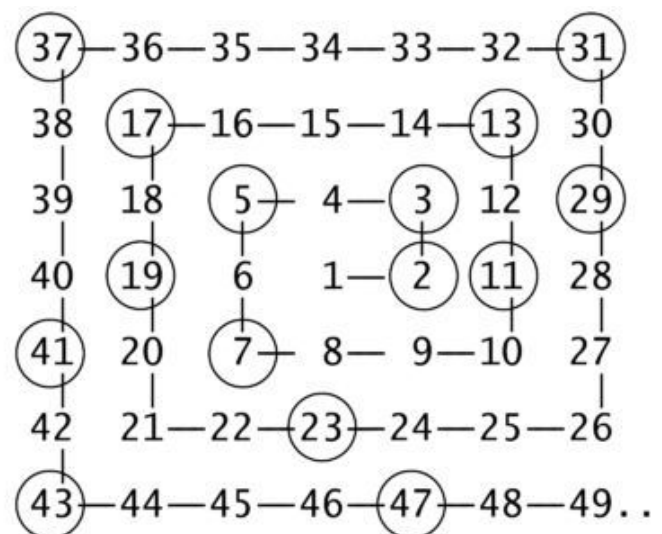
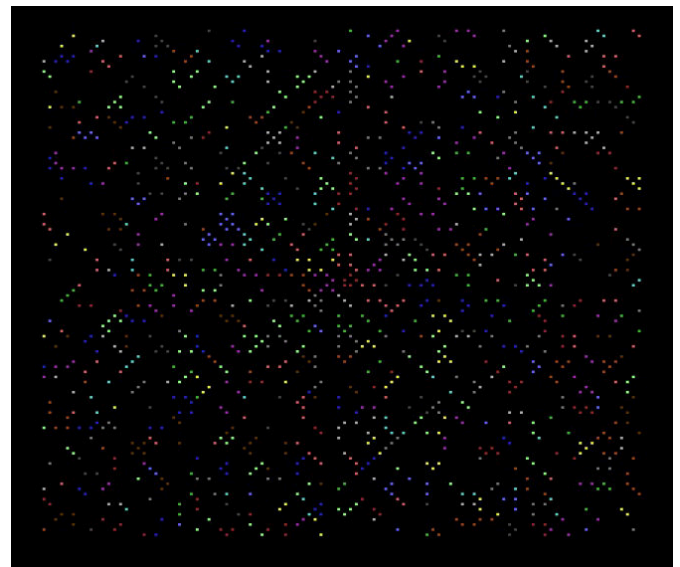
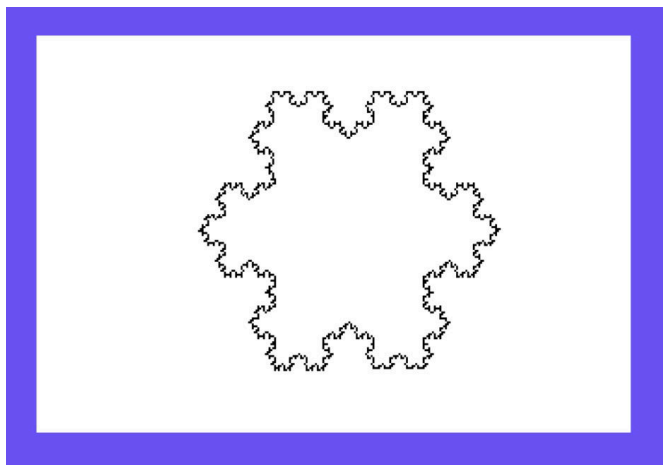


Figura A - La spirale di Ulam, fu scoperta dal matematico polacco Stanislaw Ulam nel 1963, che notò che i numeri primi tendevano ad allinearsi lungo le diagonali della spirale.



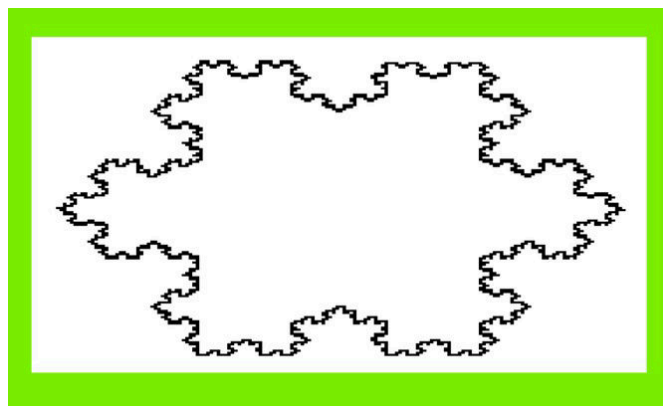


```
240 ifdv<=sqr(np) then220
250 r=1
300 return
```



18. C64 + Simons' BASIC - Il fiocco di neve di Koch (con il contributo di Armando Pavese)

```
10 hires0,1
20 dims(20)
30 a=0:d=π/3:x=128:y=168
40 fori=1to3
50 a=a+2*d
60 s(0)=140:sp=1
70 gosub100
80 next
85 wait198,1:nrm
90 end
100 sp=sp-1:l=s(sp)
110 plotx,y,1
120 ifl<=5thenh=l*sin(a):v=l*cos(a):
line x,y,x+h,y+v,1:x=x+h:y=y+v:return
130 l=l/3
140 forp=sptosp+3:s(p)=l:next
150 sp=sp+4:gosub100:a=a-d:gosub100
160 a=a+2*d:gosub100:a=a-d:gosub100
170 return
```



19. VIC-20 + SUPER EXPANDER JP VIC-1211M (6502, 1981) - Il fiocco di neve di Koch (con il contributo di Armando Pavese)

```
10 hires1:setc1,2,6
20 dims(20)
30 a=0:d=π/3:x=48:y=148
40 fori=1to3
50 a=a+2*d
60 s(0)=140:sp=1
70 gosub100
80 next
85 wait198,1:text
90 end
100 sp=sp-1:l=s(sp)
110 plotx,y
120 ifl<=5thenh=l*sin(a):v=l*cos(a):
x=x+h:y=y+v:plottox,y:return
130 l=l/3
140 forp=sptosp+3:s(p)=l:next
150 sp=sp+4:gosub100
155 a=a-d:gosub100
160 a=a+2*d:gosub100
165 a=a-d:gosub100
170 return
```

La terza parte della rubrica si ferma qui, ma il successo che abbiamo riscontrato finora fra i lettori ci fa sperare che avremo altri interessanti listati da proporvi anche in futuro, per avvicinarvi di nuovo alla programmazione BASIC e ai vostri amati microcomputer a 8-bit.

Quindi, se avete idee o spunti interessanti per la rubrica, vi invitiamo a mandarli corredati da qualche riga di informazioni e/o istruzioni alla casella e-mail:

RetroMagazine.Redazione@gmail.com.

Tutti i contributi saranno visionati e presi nella giusta considerazione e, come sempre, i migliori ed i più intriganti otterranno l'onore della pubblicazione nei prossimi numeri della rivista.





C128 in modalità 80 colonne - parte 2

di Gianluca Girelli

Nell'articolo precedente, pubblicato su RMW #34-IT, abbiamo introdotto il chip 8563 (chiamato anche VDC) cioè quella parte del sistema dedicato alla gestione dello schermo in modalità a 80 colonne.

Abbiamo imparato che, a differenza del VIC (chip 8564) utilizzato per la modalità standard a 40 colonne, non è possibile accedere direttamente al VDC con i comandi PEEK e POKE del BASIC ma è necessario usare il linguaggio macchina e fare riferimento a due locazioni di memoria specifiche: la 54874 (address register, \$D600) e la 54875 (data register, \$D601).

Tramite esse, infatti, avremo accesso ai 37 registri del chip e la loro manipolazione ci permetterà di agire sulla sua RAM di 16KB, indipendente da quella di sistema.

Abbiamo anche visto come scrivere dei dati in memoria attraverso questi registri: bisogna inserire in 54874 il numero del registro con cui si vuole interagire e, quando i bit 6 e 7 assumeranno il valore "1" (gestiti dal sistema), si può memorizzare un valore in 54875. Ciò forzerà il passaggio di tale valore nel registro puntato da 54874. Questo meccanismo ci è stato utile per riprogrammare il font caratteri di base del C128 ed ottenerne uno "Japanese-like". Il codice associato è stato commentato nel precitato numero 34, al quale si rimanda per ulteriori informazioni.

Il funzionamento delle locazioni dedicate all'I/O è lo stesso anche nel caso dell'operazione di lettura dei registri dell'8563: semplicemente, quando i bit 6 e 7 vanno ad "1", basta leggere il contenuto di 54875 che conterrà il valore del registro desiderato e "puntato" dalla locazione 54874.

Vediamo come.

Il codice che segue, preso dal libro "Commodore 128 Oltre il Manuale", fa uso della funzione BASIC "USR".

Tale comando, impiegato per implementare una funzione definita dell'utente ampliando il set di routine del C128, una volta chiamato con una sintassi del tipo "W=USR(K)" si comporta come segue:

- pone il valore di "K" nell'accumulatore in virgola mobile numero 1 (denominato "FAC1");
- trasferisce il controllo ad una routine in linguaggio macchina scritta dall'utente e memorizzata a partire

dall'indirizzo definito tramite le locazioni (decimali) 4633 (byte basso) e 4634 (byte alto) [Nota: sul C64 le locazioni hanno indirizzo decimale 785 e 786.];

- una volta che la routine LM è stata eseguita, essa pone il risultato in FAC1 prima di tornare al programma chiamante;
- il programma chiamante pone il risultato nella variabile "W" (nel caso dell'esempio) o la stampa direttamente a video nel caso si utilizzi un'istruzione del tipo "PRINT USR(K)".

Iniziamo innanzitutto a riportare il codice completo della routine in linguaggio macchina, la cui prima parte (relativa alla scrittura dei registri) è stata commentata sul numero 34:

```
F0B00 LDY #$00
F0B02 STY $FF00
F0B05 STA $D600
F0B08 BIT $D600
F0B0B BPL $0B08
F0B0D STX $D601
F0B10 RTS
F0B11 LDA #$00
F0B13 STA $FF00
F0B16 JSR $AF0C
F0B19 LDA $16
F0B1B STA $D600
F0B1E BIT $D600
F0B21 BPL $0B1E
F0B23 LDY $D601
F0B26 LDA #$00
F0B28 JMP $AF03
```

Come riportato nell'articolo precedente, abbiamo scelto di memorizzare la routine nel buffer di cassetta, che inizia alla locazione 2816, (esadecimale 0B00). Per chi non fosse pratico del C128, riportiamo la procedura per inserire programmi in memoria usando il "monitor" del linguaggio macchina:

- digitare il comando BASIC "MONITOR";
- inserire la prima riga digitando "A F0B00 LDY #\$00", seguito da "RETURN". L'editor tradurrà il codice direttamente in linguaggio macchina calcolando l'occupazione di memoria in byte e inserendo il corretto numero di riga successivo.

Si noti che l'indirizzo "F0B00" indica che stiamo scrivendo





in memoria nel "banco 15", "F" in esadecimale (consultare la bibliografia per ulteriori informazioni);

- una volta inserito tutto il codice, premere "RETURN" per uscire dalla modalità di editing, salvare il programma con il comando `S "RR8563" 8 F0B00 F0B2B` poi uscire dal monitor premendo "X" (RETURN). Si noti che il l'ultimo byte del codice viene inserito nella locazione F0B2A, ma per memorizzarla in un file binario si deve aumentare tale indirizzo di 1 byte.

Tralasciando ciò che succede nel programma tra le locazioni F0B00 e F0B10, già esaminate in precedenza, cerchiamo di capire come funziona il processo di lettura supponendo di voler leggere il contenuto del registro numero 3 dell'8563:

- prendiamo l'indirizzo di inizio della sottoroutine di lettura (OB11, il numero di banco non è importante in questo caso) e lo separiamo nelle sue componenti "OB" (decimale 11) e "11" (decimale 17). Questi valori, nella forma byte basso e byte alto, vanno settati con POKE 4633,17 e POKE 4634,11 per indicare alla routine "USR" dove saltare;

- chiamiamo la routine con il comando "USR(3)". Il valore "3" (è il registro da cui vogliamo leggere) verrà inserito nell'accumulatore FAC1;

- a questo punto il programma LM invoca la routine di sistema "GETADR" (JSR \$AFOC), che converte il valore di FAC1 in un intero di due byte. Il byte basso viene memorizzato nella locazione decimale 22 (\$16), quello alto in quella successiva.

A noi interessa solo la prima in quanto rappresenta il numero di registro con cui stiamo interagendo, quindi l'istruzione "LDA \$16" preleva il numero di registro (3) e lo mette nella locazione di I/O \$D600;

- una volta che i bit 6 e 7 di \$D600 passano a 1, possiamo automaticamente leggere il contenuto del registro "3" prelevandolo dalla locazione successiva. Tale valore verrà caricato nel registro del chip 6502 denominato "Y";

- dopo aver caricato l'accumulatore con il valore "0" (LDA #\$00) per ricostruire un indirizzo di due byte, invochiamo la routine di sistema "GIVAYF" (JMP \$AFO3). Essa si occupa di trasformare il numero di due byte che trova appunto in "A" e "Y" ("A" contiene il byte alto ed "Y" quello basso) in un valore da re-immettere in FAC1, da cui viene estratto assegnandolo ad una variabile (ad esempio con la pre-citata sintassi tipo "W=USR(3)") oppure stampandolo a video, come possiamo vedere nel codice BASIC seguente.

```
1 rem reg8563
3 rem scrittura nei registri del chip 80
  colonne
```

```
7 bload"rr8563",d0,u8,b15,p2816
13 poke 4633,17:poke 4634,11
15 print"contenuto registri 8563":print
17 for k=0 to 36
19 : print"registro ",k,usr(k)
21 next k:print
25 print"vuoi scrivere nei registri"
27 input"del chip 8563? (s/n) ";r$
29 if r$="n" then stop
31 input"scrivere nel registro (0/36) ";n
33 input"il dato: ";d
37 sys 2816,n,d
39 goto 25
```

Il programma si occupa di caricare in memoria la routine LM precedente digitata e salvata attraverso il "MONITOR" del linguaggio macchina (riga 7). Successivamente, dopo aver impostato i puntatori alla routine (riga 13), viene stampato a video il contenuto di tutti i registri del chip ad 80 colonne. Infine, il codice ci permette, se vogliamo, di scrivere dei valori in un registro a scelta e verificare cosa succede.

Ora che abbiamo imparato a leggere e scrivere nella memoria del chip, abbiamo davanti a noi tutta una nuova serie di possibilità, tra cui quella di veloci trasferimenti di blocchi di memoria o quella di fare grafica (monocromatica) a 80 colonne. Come probabilmente già saprete, le primitive grafiche del BASIC del C128 funzionano solo in modalità a 40 colonne, ma ora che abbiamo capito come accedere allo schermo del VDC possiamo scrivere le nostre routine con cui divertirci.

Appuntamento quindi al prossimo numero di RMW per esplorare nuove opportunità!

Bibliografia:

"Commodore 128 Oltre il Manuale", Rita Bonelli, Gruppo Editoriale Jackson Giugno 1986
 "C128 Programmer's reference guide", Commodore Business Machines, Inc., Bantam Book February 1986
 "Mapping the Commodore 128", Ottis R. Cowper, COMPUTE! Publications, Inc 1986
 "Commodore 128 - Manuale di Sistema", Commodore Italiana S.p.A., 1985





Come creare un gioco in BASIC per Commodore 64 - parte 1

di Felice Nardella (aka Kimono)

Questo tutorial è rivolto a chi, come me, sin da bambino, ha sempre sognato di creare un videogioco per il mitico Commodore 64, ma non l'ha mai potuto realizzare per mancanza di conoscenze sufficienti sul BASIC e soprattutto di nozioni, almeno basilari, sulla programmazione di grafica e suoni del C64 stesso, di cui mi occuperò maggiormente in questo Tutorial.

Il linguaggio BASIC v2, notoriamente integrato nella macchina, è infatti piuttosto ostico, soprattutto quando si vuole programmare con sprite e/o effetti sonori, vista la totale mancanza di comandi specifici, rispetto ad altre versioni più evolute di Basic.

La totale mancanza di questi comandi fa sì che occorra, giocoforza, utilizzare direttamente le locazioni di memoria attraverso l'istruzione POKE, introducendo il valore appropriato nei vari registri, che pure bisogna conoscere in modo più o meno approfondito, per riuscire a programmare un gioco che sia quantomeno decente. Un altro grande handicap del Basic è la sua risaputa lentezza rispetto al linguaggio Assembly (ASM); lentezza dovuta principalmente al fatto che le varie istruzioni e i vari comandi Basic devono essere prima tradotti dal cosiddetto Interprete Basic, in linguaggio macchina, linguaggio che è, per l'appunto, l'unico direttamente comprensibile dalla macchina.

Nonostante tutto, alla lentezza del Basic, nei casi in cui se ne abbia bisogno, si può porre rimedio in due modi:

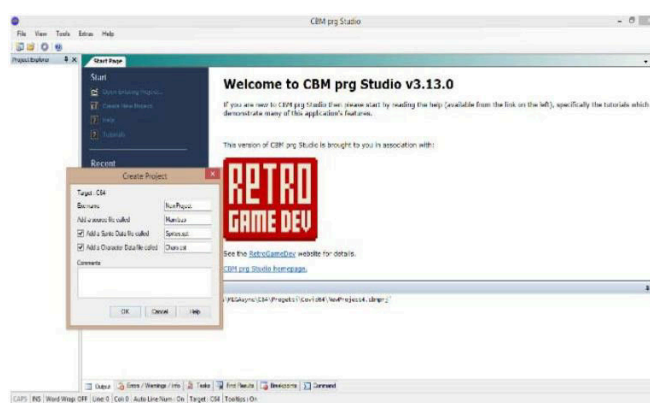
- 1) Cercare di ottimizzare il codice Basic il più possibile, seguendo delle regole di massima che descriverò nel corso di questo Tutorial, in modo da rendere più veloce la sua esecuzione.

- 2) Compilare il codice Basic mediante appositi compilatori, che, altro non sono che programmi che cercano di tradurre tutto il codice Basic, direttamente in codice macchina.

In questo Tutorial affronteremo quindi, tutti gli argomenti principali per poter creare un gioco in Basic v2 per C64 ed infine ne creeremo uno, ex novo, come esempio. Che cosa serve per cominciare Negli anni '80 ci si serviva



direttamente del C64 per scrivere programmi e giochi, oggi invece, con l'avanzare degli anni e della tecnologia, quando si voglia scrivere programmi di una certa lunghezza, la scelta è quasi obbligata all'utilizzo di IDE moderni, come il famoso CBM prg Studio, che potrete scaricare da questo link: <https://www.ajordison.co.uk/download.html> e che faciliterà non poco il nostro lavoro nella programmazione del gioco.

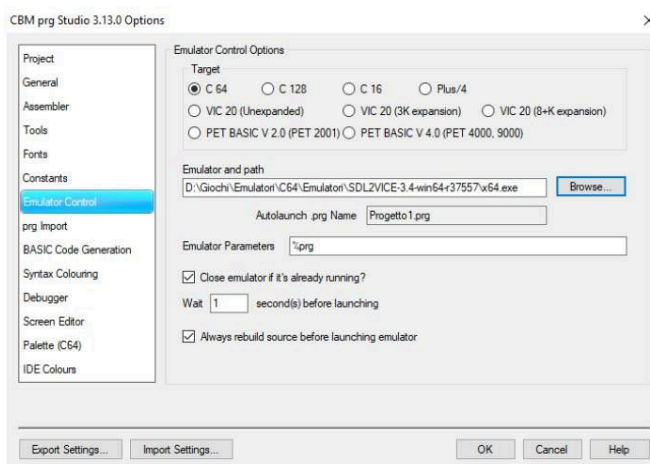


In alternativa se ne potranno usare anche altri, come ad esempio il C64 Studio: <https://www.georgrottensteiner.de/en/index.html> anche se nel corso di questo Tutorial mi occuperò solamente del primo (CBM prg Studio).

CREARE UN NUOVO PROGETTO BASIC SU CBM PRG STUDIO

Dopo aver scaricato l'IDE "CBM prg Studio" e dopo averlo installato ed avviato per la prima volta, occorre settare alcune opzioni importanti, tra cui la directory di default in cui salvare i nostri nuovi progetti. Inoltre è possibile collegare il nostro emulatore di C64 all'IDE, andando su

TOOLS -> Options... -> Emulator Control



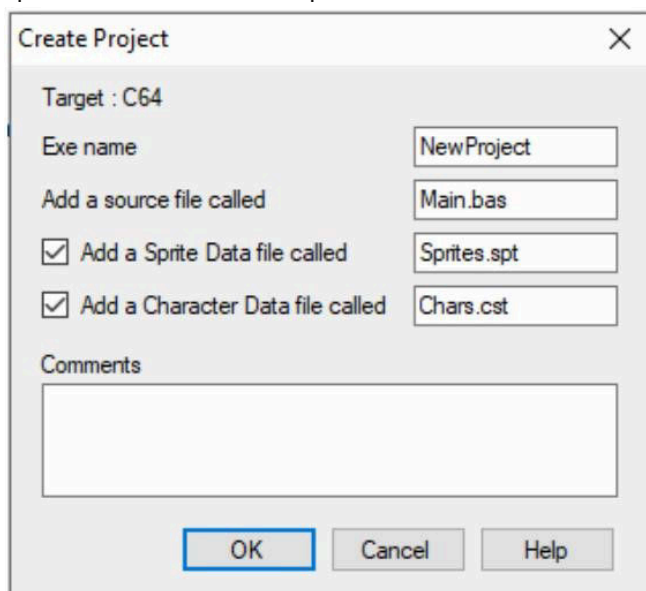


ed inserendo il percorso completo, in cui si trova il file eseguibile (exe) dell'emulatore, nel campo “**Emulator and path**”.

L'emulatore che utilizzo normalmente è il VICE; se non lo avete, potete scaricarlo da qui:

<https://viceemu.sourceforge.io/index.html#download>

Dopodiché andate su **FILE -> New BASIC Project**; vi si aprirà una finestra come questa:



Nel campo “**Exe Name**” dovremo inserire il nome del nostro progetto ad es. “**Progetto1**”; nel campo sottostante occorre invece inserire il nome del file in Basic che poi, una volta terminato, verrà compilato, creando così il file .prg che sarà possibile eseguire direttamente su emulatore.

Le due spunte sottostanti possiamo lasciarle: la prima serve ad aggiungere la possibilità di inserire uno o più Sprite, l'altra per modificare i caratteri del C64.

Quindi premiamo OK: in tal modo abbiamo creato un nuovo Progetto Basic e lo abbiamo personalizzato con un nuovo nome del progetto e del file che conterrà il nostro programma in Basic.

COSA C'È E COSA NON C'È

In questo Tutorial, come d'altronde è specificato nel titolo, mi occuperò essenzialmente della programmazione di grafica e suoni col Basic v2, al fine di scrivere un gioco completo.

Non mi occuperò invece, delle basi della programmazione in Basic v2, per le quali si rimanda ai vari manuali in circolazione sul Web.

CAPITOLO 1:

LA MEMORIA DELLO SCHERMO (SCREEN MEMORY)

Lo schermo del C64 è essenzialmente un rettangolo costituito da 25 righe (da 0 a 24) e 40 colonne (da 0 a 39). È possibile suddividere questo rettangolo in 1000 (25 x 40) caselle (vedi Fig. 1) in ognuna delle quali è

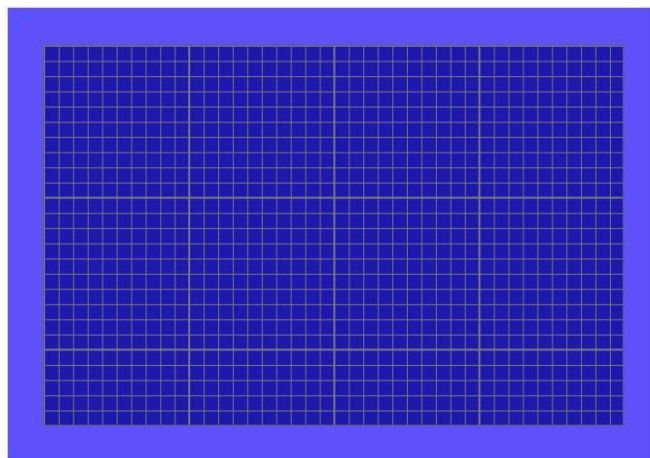


Fig. 1

possibile collocare un qualsiasi carattere PetSCII, sia esso una lettera, un numero, o un simbolo grafico. Ad ogni casella corrisponde un indirizzo di memoria secondo lo schema indicato nella fig. 2 (pagina seguente).

La numerazione comincia con la casella nell'angolo superiore sinistro (**1024**) e termina con la casella dell'angolo inferiore destro (**2023**). Si noti che le caselle appartenenti ad una stessa colonna, differiscono tra loro di 40 unità. Tutti i caratteri della tastiera invece, corrispondono ciascuno ad un numero compreso fra 0 e 127. Aggiungendo 128 ai numeri suddetti, si hanno gli stessi caratteri, ma in **reverse**.

Per poter vedere tutti i caratteri ed i codici (Screen Code) ad essi corrispondenti, è possibile usare CBM prg Studio, cliccando sul Menu **TOOLS -> SCREEN EDITOR**, si aprirà una schermata simile a quella in figura 3.

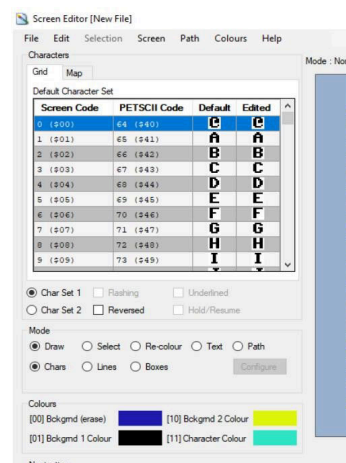


Fig. 3

Se ora, ad esempio, volessimo far comparire uno qualsiasi dei caratteri PetSCII sullo schermo, potremo utilizzare l'istruzione: **POKE A, B**. Dove A indica l'indirizzo corrispondente alla casella del rettangolo e B rappresenta il codice del carattere corrispondente.





COL.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39		
RIGHE	0	1024																																								1063
1	1064																																									1103
2	1104																																									1143
3	1144																																									1183
4	1184																																									1223
5	1224																																									1263
6	1264																																									1303
7	1304																																									1343
8	1344																																									1383
9	1384																																									1423
10	1424																																									1463
11	1464																																									1503
12	1504																																									1543
13	1544																																									1583
14	1584																																									1623
15	1624																																									1663
16	1664																																									1703
17	1704																																									1743
18	1744																																									1783
19	1784																																									1823
20	1824																																									1863
21	1864																																									1903
22	1904																																									1943
23	1944																																									1983
24	1984																																									2023

Fig. 2

Se ad esempio scrivessimo: **POKE 2023,0**

comparirebbe una chiocciola @ (carattere corrispondente a 0) nell'angolo inferiore destro (Fig. 4); mentre se scrivessimo: **POKE 1024,128** comparirebbe una chiocciola in reverse (0+128) nell'angolo superiore sinistro (Fig. 5).



Fig. 4



Fig. 5

In un'altra regione della RAM esistono altri indirizzi che immagazzinano il colore con cui deve essere visualizzato

il carattere in ciascuna casella dello schermo.

Questi indirizzi vanno da **55296** a **56295**; quindi se ad esempio volessimo colorare la chiocciola in basso a destra di nero dovremmo scrivere: **POKE 56295,0**. Dove lo 0 rappresenta il codice del colore nero.

Tutti i codici dei colori possono essere desunti dai vari manuali, oppure anche direttamente da CBM prg Studio cliccando sul **Menu TOOLS -> SCREEN EDITOR** e cliccando poi 2 volte su uno dei 4 colori presenti in Fig. 3; verrà fuori una piccola finestra come quella mostrata in Fig. 6



Fig. 6

Tramite una semplice formuletta, saremo in grado di collocare un qualsiasi carattere in un punto qualsiasi dello schermo, sapendo solo la riga e la colonna.





L'istruzione con la formula generale è la seguente:

POKE 1024+RIGA*40+COLONNA,CARATTERE

Mentre per colorarlo dovremmo usare quest'altra formula:

POKE 55296+RIGA*40+COLONNA,COLORE

Ad esempio se volessimo far comparire un cuoricino rosso alla riga 13 e alla colonna 0 (si ricordi che la colonna 0 è la prima colonna a sinistra) dovremo scrivere un programmino di sole 2 righe e dare RUN:

10 POKE 1024+13*40+0,83

20 POKE 55296+13*40+0,2



Trick: per scrivere "al volo" un programma piccolo come quello appena visto, ma anche di dimensioni maggiori, senza passare per l'IDE, di solito non conviene scriverlo direttamente su emulatore. Conviene, invece, scriverlo (tutto in minuscolo) in un file di testo (txt) e successivamente fare il copia/incolla sull'emulatore.

UN PRIMO ESEMPIO: UNA PALLINA RIMBALZANTE

Ora che abbiamo capito come visualizzare i caratteri a video, proviamo a battere su CBM prg Studio un programmino che visualizzi una pallina rimbalzante su schermo. Creiamo su CBM prg Studio un nuovo Progetto Basic e chiamiamolo "Esempio1". Poi rinominiamo il file "Main.bas" in "Pallina.bas". In questo caso possiamo togliere le due spunte sottostanti, in quanto in questo primo esempio non avremo né la necessità di usare sprite, né quella di cambiare i caratteri standard del C64. Ora digitiamo il seguente programma:

```
10 print"{clear}": rem cancella lo schermo
20 sm=1024:cm=55296: rem inizializza le variabili
   screen memory e colour memory
30 p=81: rem memorizza il cod. del carattere
   della pallina
40 ko=7: rem memorizza il codice del colore della
   pallina (giallo)
```

```
50 ca=32: rem memorizza il cod. del car. spazio
   usato per cancellare la pallina
60 co=1:ri=1: rem inizializza la posizione
   iniziale della pallina
70 dx=1:dy=1: rem inizializza l'incremento
   orizzontale e verticale
80 pp=ri*40+co: rem calcola la posizione iniziale
   della pallina
90 co=co+dx:ri=ri+dy:np=ri*40+co: rem aggiorna
   la posizione della pallina
100 if ri>23 then dy=-1: rem controlla che la
   pallina non scenda oltre la riga 24
110 if ri<1 then dy=1: rem controlla che la
   pallina non salga oltre la riga 0
120 if co>38 then dx=-1: rem controlla che la
   pallina non vada oltre la colonna 40
130 if co<1 then dx=1: rem controlla che la
   pallina non vada oltre la colonna 0
140 poke sm+np,p: poke cm+np,ko: rem stampa a
   video la pallina e la colora
150 poke sm+pp,ca: pp=np: rem cancella la pallina
   e aggiorna la pos. iniziale
160 goto 90: rem ricomincia il ciclo dalla riga 90
```

Come potete osservare ogni riga è stata debitamente commentata (i commenti possono anche essere omessi nel ricopiare il codice, basta che li leggete e capiate cosa faccia, riga per riga, il programma). Dopodiché premete sul bottone PRG in alto. Se non sono stati commessi errori, dovrete trovare il vostro file PRG già pronto per essere eseguito all'interno della cartella che avete scelto come default per i progetti. Se avete già collegato il VICE all'IDE tramite le impostazioni, vi basterà premere sul pulsante immediatamente a destra (->) per mandarlo direttamente in esecuzione sul VICE.





Musica Maestro SID!

di Felice Nardella (aka Kimono)

Sappiamo tutti che un file SID non può essere ascoltato tal quale su C64 reale o su emulatore... Ma se invece voleste ascoltarlo, come fareste?

Beh, ci sono diversi modi per farlo: uno piuttosto semplice è quello di trasformare direttamente il SID in un file PRG eseguibile, tramite il tool PSID64:

<http://psid64.sourceforge.net/>.

Ma noi siamo abituati a sporcarci le mani con il codice, quindi vi presenterò qui un programmino, che servirà al nostro scopo.

È bene precisare che non è possibile effettuare tale operazione da Basic, ma occorre utilizzare l'Assembly; in questo caso ho usato il Basic solo per caricare il codice ASM (e il SID) in memoria ed eseguirlo. Questo metodo funzionerà con quasi tutti i file SID (scaricabili, ad esempio, dal sito <https://hvsc.c64.org/>), ma non con tutti.



In particolare, funzioneranno soprattutto i SID che hanno un INIT ADDRESS alla locazione \$1000 (4096 dec). Se invece il SID avrà un indirizzo diverso, questo metodo non funzionerà, ma non tutto è perduto! Esiste un tool chiamato SIDRELOC, scaricabile da un sito molto ben fatto, dove si possono attingere varie informazioni utili sui SID (<http://www.linusakesson.net/software/sidreloc/index.php>), che è in grado di rilocare il SID nell'area di memoria voluta (una versione di SIDRELOC compilata ed eseguibile su



Windows, la potrete scaricare da CSDB: <https://csdb.dk/release/?id=109000>).

Nello specifico, qui ho cercato di eseguire il file SID del famoso gioco OUTRUN, che potrete trovare qui: <https://csdb.dk/sid/?id=4154>.

Eseguendo tale file, tramite il tool Sidplayfp (<https://csdb.dk/release/?id=210057>), occorre andare a guardare tra le proprietà del file (attraverso il menu FILE -> PROPERTIES), soffermandoci soprattutto sui seguenti valori:

- Init address: **\$B219**
- Play address: **\$B069**
- Number of songs: **2**

Ci si accorge dunque, che questo SID ha un Init address diverso da \$1000 e che, inoltre, possiede 2 tracce.

Quindi dobbiamo cercare di rilocarlo il più vicino possibile all'indirizzo \$1000, mediante il già citato SIDRELOC, andando a digitare da riga di comando:

sidreloc -p 10 out_run.sid outrun.sid

```
C:\Users\...>sidreloc -p 10 C64\SID\sidreloc-1.0-win32\rem sidreloc
C:\Users\...>sidreloc -p 10 C64\SID\sidreloc-1.0-win32\sidreloc -p 10 out_run.sid outrun.sid
Out Run, Jason C. Brooke, 1987 US Gold, $5000-$22ff, 2 subtones
Relocating from $0000-$22ff to $1000-$22ff
Analyzing subtone 1
Analyzing subtone 2
Verifying relocated subtone 1
Verifying relocated subtone 2
Bad pitches: 272, 0%
Bad pulse widths: 0, 0%
Relocation successful with some mismatching pitches.
```

In questo caso gli errori di "Bad pitches" possono essere tranquillamente ignorati.

Osservando ora tra le proprietà del nuovo file ottenuto (outrun.sid), avremo i seguenti parametri:

SID Tune Properties

```
Filename: C:\Users\...>sidreloc -p 10 C64\SID\outrun.sid
Data size: 4864 bytes
File type: PlaySID one-file format (PSID)
File status: No errors
Name: Out Run
Author: Jason C. Brooke
Released: 1987 US Gold
Load range: $1000-$22FF
Init address: $1219
Play address: $1069
Number of songs: 2
Default song: 2
Clock speed: Unknown
SID model: Unknown
Player routine: Jason_Brooke
Reloc region: $2300-$9FFF
```

Emulation

```
-----
SID model: Unknown
Clock speed: 50 Hz VBI (PAL)
Environment: Real C64
Driver region: $2300-$23FF
Power-on delay: 3463
```

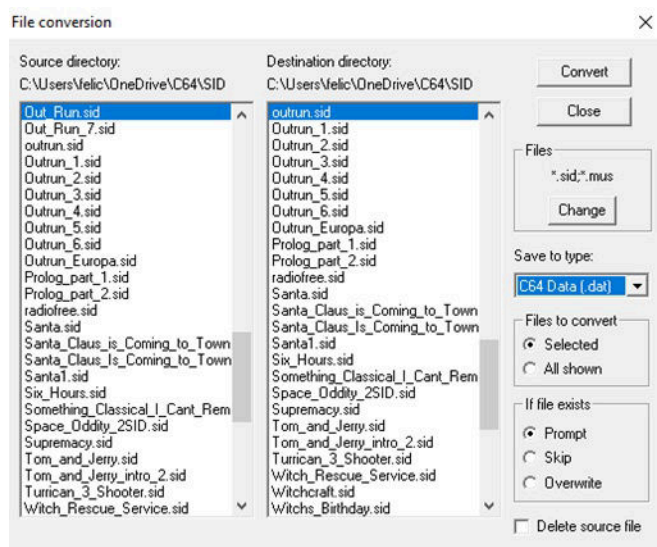




- Init address: **\$1219**
- Play address: **\$1069**

Questi due valori dovranno essere inseriti nel nostro codice.

Prima di eseguire il Sid, però, occorre preventivamente trasformarlo in un file .DAT, sempre tramite il tool Sidplayfp, andando su FILE -> CONVERSION, selezionando su "Save Type" l'opzione "C64 Data (.dat)" e successivamente cliccando su CONVERT. Otterremo così, il file "outrun.dat" che andremo ad utilizzare più avanti. Questa trasformazione è necessaria in quanto i file SID hanno un'intestazione iniziale di lunghezza \$7c (124 dec) byte che, in questo caso, è necessario rimuovere prima di poterli posizionare nella memoria del C64.



È anche possibile eliminare "a mano" questa parte, usando un buon editor in esadecimale, come ad esempio WINHEX (<http://www.winhex.com/winhex/>), vedi fig. 1.

Il codice che ho usato è il seguente:

```
10 poke53280,0:poke53281,0:me=49152
20 readop:ifop<0then90
30 pokeme,op:me=me+1
```

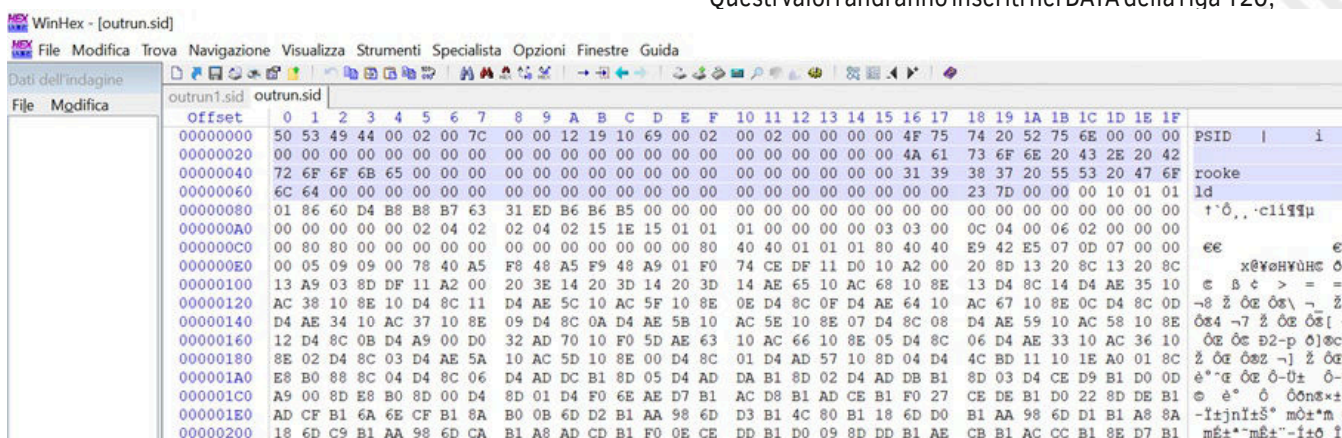


Fig. 1 - Editor esadecimale WINHEX

```
40 goto20
90 ifa=0thena=1:load"outrun.dat",8,1
95 poke53280,15:sys49152
100 data 120: rem sei
110 data 169,0: rem lda #0
120 data 32,25,18: rem jsr $1219 - init addr
130 data 32,105,16: rem jsr $1069 - play addr
140 data 173,18,208: rem lda $d012
150 data 201,250: rem cmp #250
160 data 208,249: rem bne 1140
170 data 173,1,220: rem lda $dc01
180 data 201,255: rem cmp #$ff
190 data 240,239: rem beq 1130
200 data 169,0: rem lda #0
210 data 141,24,212: rem sta $d418
220 data 88: rem cli
230 data 96: rem rts
240 data -1
```

Ricordiamoci, intanto, il nome del file DAT appena ottenuto, che dovremo andare a scrivere nella riga 90 (come indicato nel listato).

Nei DATA delle righe 120 e 130 dovremo andare ad inserire, rispettivamente i valori di Init Address e Play Address. Per chi fosse in difficoltà nell'eseguire tale passaggio, di seguito spiegherò brevemente come procedere:

- il valore \$1219 (in esadecimale) viene espresso in binario così:

0001 0010 0001 1001

questo è un valore a 16 bit, che occorre scomporre in due valori ad 8 bit.

- Il primo valore da inserire è rappresentato dal byte basso (a destra) che, trasformato in decimale è:

0001 1001 = 25

- Il secondo è dato dal byte alto (a sinistra):

0001 0010 = 18

- Questi valori andranno inseriti nei DATA della riga 120,





dopo il valore 32 (che rappresenta l'OpCode JSR in decimale).

- La stessa operazione andrà effettuata per i valori del Play Address.

Per cambiare traccia, nel caso ce ne fossero più di una (nel nostro caso ne abbiamo 2), basta andare a cambiare il valore nella riga 110 dopo la cifra 169 (OpCode LDA in decimale), in questo caso da 0 a 1, oppure è anche possibile cambiare direttamente il valore della locazione 49153 tramite una POKE:

POKE 49153,1

e rieseguire il codice con **RUN**.

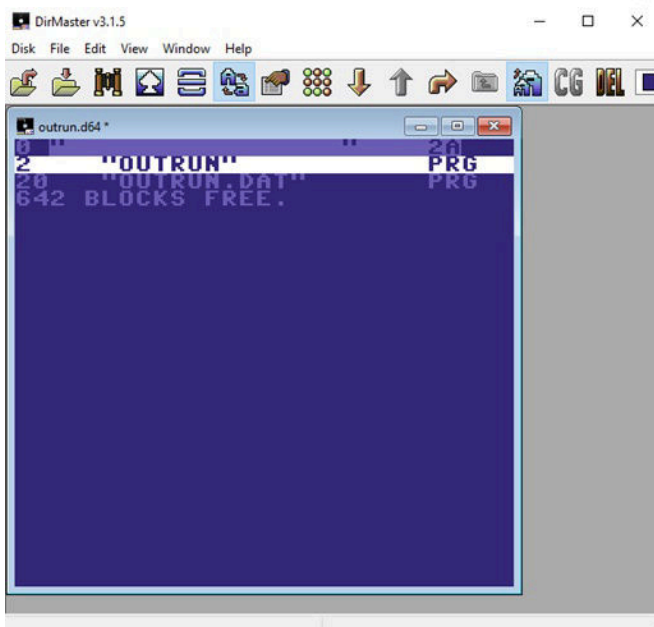
La locazione \$DC01 (56321 dec) indicata nella riga 170, è quella che contiene lo stato del joystick in porta 1, ma anche di alcuni tasti della tastiera (in particolare SPAZIO, CTRL, 1, 2, FRECCIA). Se uno di questi tasti viene premuto durante l'esecuzione, oppure se viene mosso il joystick in porta 1, il programma esce. Si sarebbe potuto immettere anche un'altra locazione da controllare, come ad esempio quella del joystick in porta 2 (\$DC00); in quel caso i tasti da premere per uscire dal programma, sarebbero stati SPAZIO + F1/F2, SPAZIO + Z, SPAZIO + B, SPAZIO + C, SPAZIO + M, oppure il movimento del joystick in porta 2. Il valore nella riga 180 (di controllo) sarebbe dovuto cambiare da 255 a 127 (\$7F).

Le righe 200 e 210 prima di uscire, azzerano il volume del SID, agendo sul Registro \$D418 del chip audio, mentre l'ultimo DATA della riga 240 (-1) è semplicemente un byte di controllo per l'uscita dal caricamento della routine Assembly in memoria, a partire dalla locazione 49152 (\$C000).

Il file PRG così ottenuto dovrà essere racchiuso in un file D64 insieme al file DAT, tramite l'utilissimo tool DIRMASTER (<https://style64.org/dirmaster>). Il file PRG, all'interno del D64, dovrà necessariamente precedere il file DAT, se si vuole che venga avviato automaticamente dall'emulatore.

Il file D64 che ne deriva potrà, poi, essere agevolmente eseguito su emulatore o copiato su floppy (o altro supporto) per essere avviato da C64 reale.

Buon ascolto!



Link utili

Riportiamo qui di seguito i link presentati nell'articolo per una migliore fruibilità degli stessi.

PSID64:

- <http://psid64.sourceforge.net/>

HVSC - High Voltage SID Collection:

- <https://hvsc.c64.org/>

SIDRELOC:

- <http://www.linusakesson.net/software/sidreloc/index.php>

SIDRELOC eseguibile in Windows:

- <https://csdb.dk/release/?id=109000>

Sidplayfp:

- <https://csdb.dk/release/?id=210057>

DIRMASTER:

- <https://style64.org/dirmaster>

SID del gioco OUTRUN:

- <https://csdb.dk/sid/?id=4154>

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

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

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





Intervista a Bas Scheijde, il papà di Puzzle Bobble su Commodore 64

di Carlo Nithaiah Del Mar Pirazzini

Puzzle Bobble in versione C64 è una perla! La recensione parla chiaro e ci teniamo a specificare che per realizzare un software di tale livello occorre conoscenza, conoscenza e... conoscenza.

Abbiamo fatto quattro chiacchiere con **Bas Scheijde**, sviluppatore e mente creativa dietro al gioco.

Nith: *Ciao Bas, Grazie per il tuo tempo. Raccontaci qualcosa di te. Quando è iniziato il tuo lavoro come retro-sviluppatore?*

Sono uno sviluppatore di software autodidatta, le mie capacità di programmazione sono principalmente 6502 assembly, Basic, VB.net e qualche piccola conoscenza di Javascript, C# e C++

La mia prima programmazione è iniziata all'età di 9 anni con un Commodore PET base, ma presto abbiamo avuto un C64 a casa.

Ad un certo punto sono stato coinvolto in un progetto in Assembler, ma soltanto riguardo ad alcuni trainer e una manciata di demo che non sono mai state pubblicate. O meglio, il gruppo di cui facevo parte, non ha mai pubblicato nulla di significativo.

Dopo il C64 è arrivato l'Amiga, ma non l'ho mai programmato, questo era puramente per giocare e fare musica.

Ma ho sempre tenuto il mio C64 nell'armadio per il futuro. Futuro che poi è arrivato 5 anni fa. Ho rispolverato il mio C64, ma ho scoperto che tutti i dischi erano spariti, il C64 dava una schermata nera e l'unità non aveva più un alimentatore.

Sono andato su un sito di mercato di seconda mano e mi sono comprato un biscottone nuovo a un prezzo molto basso, insieme a un dispositivo sd2iec e mi sono rimesso a lavorare.

A questo punto stavo anche iniziando a creare dei piccoli programmi per provare a vedere se i dispositivi touch (tablet e telefoni) potevano essere collegati all'ingresso della penna ottica e magari impostare qualcosa come una bella interfaccia grafica.

Dato che ci sono già dei bei progetti che stanno lavorando su questo, ho deciso di controllare quali giochi sarebbero ancora interessanti da portare.



**Fig. 1 - Bas in versione cuoco dopo aver cucinato...
...la torta non ha scampo**

È così che sono finito su Puzzle Bobble. Non c'era ancora un port realizzato e quello che ho potuto trovare, come Monster Buster, era sì carino, ma non un vero port dell'originale.

Quindi, ecco il mio progetto, creare un port di Puzzle Bobble al 100%.

Un altro concetto che si può ricavare dal progetto attuale è la versione SNES 100 level. Il gioco è quasi lo stesso per quanto riguarda la fisica del gioco, solo più livelli e alcune palline "speciali" che hanno caratteristiche specifiche quando vengono colpite.

Ma questo è per dopo...

Nel frattempo mi chiedevo anche se fosse possibile creare un vero e proprio pseudo 3D racer per il C64 con un framerate accettabile, utilizzando diverse tecniche.

Sono un grande fan di Lotus Esprit Turbo Challenge su Amiga e ho pensato che un tale motore potesse essere possibile per il C64.

Quindi un anno di sviluppo di Puzzle Bobble è stato interrotto per trasformare l'idea in una prova di concetto, il risultato può essere trovato sul mio canale YouTube (https://www.youtube.com/channel/UCcw3-w_mmXJEm9w3utCrN6w-NdN).

Spero di continuare presto con il progetto racing perché



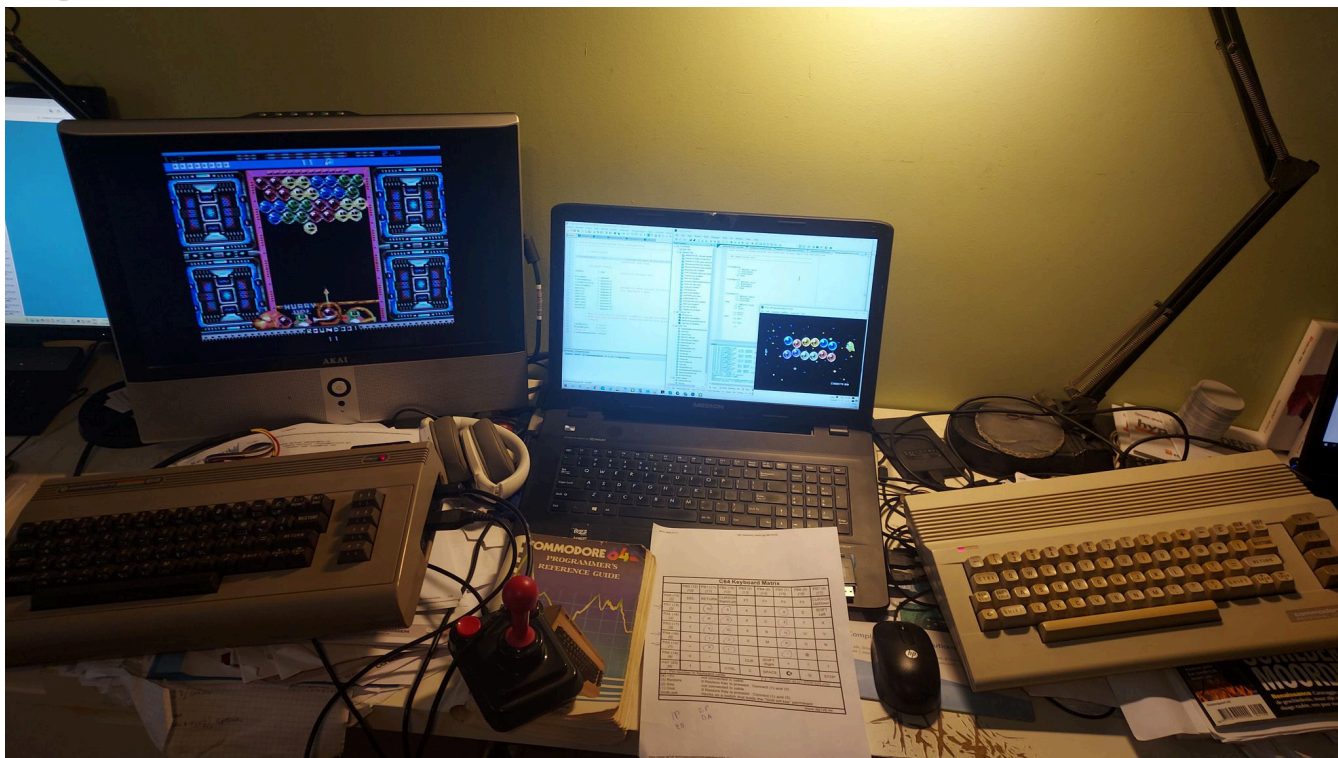


Fig. 2 - La postazione di sviluppo di Puzzle Bobble di Bas

sembra essere abbastanza buono per quanto riguarda il framerate decente e la costruzione di strade pseudo 3D completamente flessibili.

Ma è ancora molto lontano da un concetto di gioco.

N: Ora parlati di Puzzle Bobble e di come hai iniziato lo sviluppo.

Puzzle Bobble è stato creato da Taito nei primi anni '90 ed è uno dei più popolari giochi di puzzle realizzati.

Quando ho iniziato il progetto ho anche pensato che sarebbe stato divertente creare un port simile a MAME, usando le stesse chiavi e mantenendo tutto del gioco, insert coin compreso.

Ovviamente questo porta a... convertire anche l'intro. Inizialmente un compito extra piuttosto pesante, ma anche divertente da svolgere.

L'intero gioco, inclusa l'introduzione, è basato su bitmap, non vengono utilizzati set di caratteri, solo grafica bitmap e sprite. Per rendere le animazioni gestibili, ho codificato un sequencer, che può eseguire funzioni in ordine batch con determinati parametri.

Inizialmente questo è stato fatto per impostare gli sfondi dei livelli, ma in seguito è stato anche molto utile per creare le animazioni di introduzione.

Dato che ci sono così tante animazioni sullo schermo, ho anche deciso di utilizzare una sovrapposizione di sprite sul campo di gioco, in cui si svolgono alcune animazioni

come i mostriciattoli che cadono quando escono dai palloncini e i punti fluttuanti. Anche queste animazioni sono inserite in questa sovrapposizione. In 1 giocatore, 2 sovrapposizioni una sopra l'altra, in 2 giocatori, ogni giocatore ha la propria sovrapposizione, dando un leggero contrasto di colore tra punti e mostriciattoli che cadono, ma l'ho trovato accettabile.

Per gestire tutte le animazioni, utilizzo un array di animazione di 4 pagine con 255 slot, tenendo traccia ed eseguendo tutte le animazioni al di fuori dell'IRQ, mentre il gameplay stesso rimane all'interno dell'IRQ, lasciando che il gameplay non venga influenzato se si verificano molte animazioni al contemporaneamente.

Per quanto riguarda musica e sfx, ho lavorato duramente insieme a NM156 per trarne il massimo. Quindi sono entrato nel lettore GoatTracker e ho attivato direttamente i pattern con una voce libera mentre la musica suonava sulle altre 2 voci. Nel caso in cui la musica sia spenta, gli sfx sono divisi su tutte e 3 le voci, per ottenere ancora di più dagli sfx. In 2 giocatori, abbiamo dovuto accontentarci di 1 voce per la musica e ogni giocatore una voce dedicata.

NM156 (Frantic Freddy 2) ha fatto un ottimo lavoro, creando un'intera gamma di pattern per un sacco di quantità diverse di palline che scoppiano o cadono e ricreando tutti i brani musicali da zero.

Hend (Wild Woods) mi ha aiutato molto a modificare la grafica, ha aggiunto molte cose fantastiche e ha mantenuto anche alcuni dei miei progetti originali.





Alcune caratteristiche interessanti che abbiamo aggiunto con l'aiuto di Hend sono state la creazione di una stampa dedicata al livello che cambia ogni 3 livelli e il fantastico disegno dell'immagine dello schermata finale.

Tutto sommato, è stato molto tempo di sviluppo, ma sono contento che tutto abbia funzionato alla fine.

Consiglio per chiunque voglia fare progetti simili:

- creare un proof of concept giocabile, che non ha bisogno di essere bello, ma deve funzionare correttamente per la parte migliore. Puoi sempre trovare artisti musicali o grafici che ti aiutano, ma troppi progetti finiscono per diventare "lo finirò un giorno..." quando la complessità diventa troppo grande.
- non esitare a fare grandi modifiche quando noti di aver sbagliato strada nella programmazione; meglio cambiare rotta in tempo che farlo alla fine.
- crea piccoli progetti collaterali per testare cose e conoscere gli script che usi da altri. Nel mio caso ho passato molto tempo ad immergermi nella riproduzione di digi/campioni, usando Bitops Fastloader di Lasse e hackerando la routine di goattracker di Lasse per controllare la velocità di riproduzione e riprodurre i modelli al volo mentre la musica è in riproduzione.
- controlla il tuo codice frequentemente per ottimizzarne la velocità o la dimensione. Il più delle volte, il proof of concept è ottimo, ma guardare indietro ad un codice un anno dopo, cambiando la tua prospettiva, potrebbe portarti a idee per migliorare immensamente il vecchio codice.

C'è un'altra nota a margine che vorrei fare. Non avrei mai iniziato questo progetto se non fosse stato per **CBM PRG Studio** di **Arthur Jordison**. Senza un tale IDE, probabilmente non sarei stato in grado di farlo.

Tutti gli strumenti utilizzati:

CBM Prg Studio

Goattracker2

Covert Bitops Fastloader

Multipaint

Spritepad

Exomizer

Dirmaster

N: Grazie per tutte le risposte e il tuo tempo prezioso. Prima di salutarci, parlati dei tuoi progetti futuri.

Il prossimo progetto sarà il gioco **Racer**, ma in questo momento è presto per parlarne, è ancora in fase di sviluppo.



Fig. 3 - Il gioco Racer, attualmente in fase di sviluppo

Probabilmente rilasceremo anche la versione a **100 livelli** di Puzzle Bobble ad un certo punto quest'anno. (Sfida: forse qualcuno hackerà il gioco e lo fa prima?).

Grazie a te per l'intervista e alla prossima.





Intervista ad Alessandro Rogati, creatore di Salagiochi 1980

di Giorgio Balestrieri e Mic the Biker

Molti lettori, ne siamo sicuri, conosceranno molto bene Salagiochi 1980, la pagina Facebook dedicata ai videogiochi degli esordi fino al periodo '80-'90, partendo dalle sale giochi fino ad includere quelli per console e microcomputer dell'epoca. Questo progetto, partito dalla volontà del creatore di condividere la sua passione per l'intrattenimento videoludico, ha raccolto uno straordinario successo ed oggi la pagina è affiancata da canali Twitch e Youtube che insieme contano decine di migliaia di iscritti. Vi proponiamo qui l'intervista ad Alessandro Rogati, la mente dietro la pagina, che ci racconterà come tutto ciò è accaduto.

GB: Qualche parola per conoscere "l'uomo della sala giochi": chi è Alessandro nella vita non virtuale?

Sono Alessandro, per gli amici Alex, marchigiano classe '73, appassionato di videogiochi da tutta una vita praticamente. Sono un semplice impiegato in una grande azienda di abbigliamento uomo-donna e babbo di una fantastica ragazza di 22 anni.

GB: Qual è stato il tuo primo contatto con un videogioco? Quanti anni avevi, dov'è accaduto l'incontro e che gioco era?

Il primo devastante impatto con i videogiochi è avvenuto (credo) nella sala giochi del mio paesino sperduto tra le colline tra Ancona e Macerata, Filottrano, circa 10.000 anime, ad inizio anni '80. La vecchia sala giochi (dico vecchia dato che poi ne aprirà una seconda e la chiamavamo 'la sala giochi nuova') era di fianco ad uno dei due cinema paesani, e di domenica uscendo dalle proiezioni si restava letteralmente incantati dai suoni e colori che provenivano da quella stanzetta buia e maleodorante. Il primo gioco che vidi fu SPRINT di Kee Games, praticamente l'antenata di Atari, un gioco di macchine in bianco e nero fatto solo con grossi pixel, ma sul suo cabinato troneggiavano un irresistibile volante, le marce (mi sembra) e il pedale dell'acceleratore. Lascio a voi immaginare il resto...

GB: Che console e/o microcomputer hai avuto e quali avresti voluto avere?

La mia 'carriera' videoludica è molta varia e per certi versi travagliata (specialmente per i miei che dovevano cacciare la grana per acquistare le varie macchine). Si parte nei primissimi '80 con un multigioco della Philips che conteneva Pong, Calcio, Tennis, Squash, Tiro al piattello e altri, con tanto di pistola ottica che funzionava un po' a singhiozzo diciamo. Arrivò poi il mitico VIC-20, seguito a ruota dal mio mitico MSX PHILIPS VG8020. Dopo pianti e scene indicibili a casa si decisero a prendere anche il king degli home computer (ovviamente per l'epoca): il grandissimo COMMODORE 64. Nel bel mezzo di questo intreccio di fili e cassette, arrivò anche un NES, con tanti bei giochi e altrettante imprecazioni. A fine anni '90, signore e signori, arrivò lui, anzi lei, l'AMIGA 500 e dopo un annetto anche l'espansione che lo portava alla stratosferica soglia di ben 1 mega. Per qualche mese mi sono anche goduto un SUPER NINTENDO, prestato da un caro amico che non lo usava più, ma che, purtroppo, ho dovuto poi restituire altrimenti i genitori mi avrebbero linciato. La prima 'macchina' acquistata col sudore della mia fronte è stata l'indimenticabile PLAYSTATION 1, subito modificata per poter usufruire dei 'millemita' cd che si spacciavano all'epoca per poche migliaia di lire. Da lì in poi (quasi) solo ed esclusivamente PC, anche se ho comprato anche la PS2, ma non l'ho mai approfondita più di tanto.

GB: Genere e gioco preferito?

Adoro i giochi arcade, specialmente i platform, salta e spara e affini, ma ho giocato tanto a un po' tutto quello che il mercato offriva all'epoca. Come titoli arcade posso citare Wonder Boy, Rainbow Islands, Black Tiger, Psychic 5 e tantissimi altri. Se parliamo di 8, 16, 32 bit o superiori, le pagine di questa rivista non basterebbero per elencarli tutti.

GB: Oltre ai videogiochi, hai tentato anche la via della programmazione? In quali linguaggi ti sei cimentato?

Sono un 'copiatore' seriale della prima ora, quando si digitavano gli infiniti listati da PaperSoft per intenderci, ma sono da sempre una persona curiosa, e per questo





Fig. 1 - Testata della pagina facebook di salagiochi1980

ho imparato il basic rubando qua e là, seguendo i corsi a fascicoli Jackson, e leggendo i manuali delle varie macchine. Solo molti anni più tardi, dopo una lunga pausa, mi sono avvicinato all'html e php e ho iniziato a fare siti per amici e non solo. Qualche tempo dopo ho anche fatto un corso di Flash Mx sempre per lo stesso motivo, cosa che poi ho utilizzato per 'programmare' piccoli giochi in flash.

GB: Hai una tua collezione di console e microcomputer?

Negli anni è dura portarsi dietro tutti i propri gioielli specialmente se, come il sottoscritto, sei passato attraverso diversi traslochi, divorzi, convivenze e quant'altro, ma qualcosa sono riuscito a conservarla. Ho un 64, che purtroppo da un paio di mesi ha iniziato a fare le bizzarrie e non si accende più, il mio mitico MSX Philips VG8020, due AMIGA 500, di cui uno funzionante e l'altro col drive rotto, due NES con diverse cartucce al seguito, PS1 con una miriade di CD, PS2 e altro materiale. Piano piano sto cercando di ricomprare quello che vorrei conservare nella mia collezione, tipo il Vic 20, GameBoy, Master System e altro.

GB: Parliamo ora di SalaGiochi1980: com'è nata l'idea?

Dopo una vita passata sui videogiochi, inclusa anche la gestione di diversi siti e forum dedicati all'abandonware e gaming in generale, ho pensato di condividere questa mia grande passione con il resto del mondo. Il 14 Febbraio

del 2014, ho creato la pagina e scritto il primo post, il resto è storia. La pagina ha avuto un boom incredibile (nel 2014 si contavano ancora sulle dita di una mano pagine e gruppi Facebook dedicati al retrogaming), in poco tempo mi sono ritrovato con 1,000 iscritti, poi 5,000 e dopo due anni eravamo già a 45,000. Un successo che mi ha letteralmente lasciato senza parole.

GB: È nata come gruppo o l'hai creata fin da subito come pagina tematica?

SalaGiochi1980 è nata direttamente come pagina, all'epoca (parliamo del 2014) mi è sembrata la scelta migliore.

GB: Ad oggi SalaGiochi1980 ha quasi 90.000 "mi piace", quanto ci è voluto per raggiungere questo risultato?

SalaGiochi1980, come dicevo sopra, è online (su FB) dal 14 Febbraio 2014, quindi sono 8 anni, 8 anni di duro lavoro, passione, gioie, dolori e chi più ne ha più ne metta.

GB: Quando ti sei reso conto che il gioco iniziava a diventare serio?

Il mio approccio alla vita è da sempre 'semiserio', e anche in questa avventura non ho voluto cambiare forma. Le cose troppo serie alla fine stancano, a chi le fa, ma anche e a chi le guarda, vede, ascolta... Non vorrei diventasse





Fig. 2 - Alex durante una delle sue live su Twitch

un lavoro, ma sempre e solo un divertimento per passare il tempo libero, per guadagnarmi la pagnotta ho il mio impiego che già mi tiene fuori casa per 8/9 ore al giorno.

GB: Di quanti giochi hai parlato sulla pagina?

Tantissimi, ho perso il conto ormai da diverso tempo, ma conservo un mio archivio per riproporli di tanto in tanto, dato che non tutti riescono a leggere i tanti post che facciamo ogni giorno.

GB: Oltre a Facebook, sei anche su Twitch e Youtube: come riesci a trovare il tempo per fare tutto e mandare avanti la famiglia? C'è chi ti aiuta o riesci a gestire tutto da solo?

Il tempo per le proprie passioni si trova, spesso anche 'rubandolo' agli affetti e al poco tempo libero. Il primo periodo di SalaGiochi1980 è stato un cammino solitario, il mio, ma poi ho avuto al mio fianco diversi amici che mi hanno aiutato a gestire il tutto, e che non smetterò mai di ringraziare (vedi Tiz, Lighting, Meo...). Al momento un caro amico, MRAndread, mi aiuta nella pubblicazione giornaliera su FB, e mi affianca alla conduzione su diverse rubriche che facciamo in live su Twitch. Durante le serate a tema molti altri amici sono sempre pronti su Discord o in cam a dare una preziosissima mano tra cui il mitico Duncone from Malta, Andrea Pipo, EnriMassa, Marco

Turrican, Simobol, MarcoVana e moltissimi altri che non posso a citare per non rischiare di riempire un intero foglio A4. GRAZIE RAGAZZI, di vero cuore. A tutti voi!

GB: Parliamo di costi e ricavi (se ti va e se ce ne sono). Quanto costa gestire un progetto come il tuo? Ci sono forme di ricavo?

Come sopra, una passione non ha un prezzo, c'è gente che spende 100,000 dollari per un calzino usato di Michael Jackson!!! Ricavi? Bella domanda. Da FB e INSTA assolutamente nulla. Con YT e TWITCH ci paghi le bollette della fibra e qualche accessorio (cam, mic, pad...) all'anno. Se la gente pensa di fare soldi con lo streaming di retrogaming, farebbe bene a volgere lo sguardo altrove. E comunque non è assolutamente il mio scopo principale.

GB: Cosa apprezzi nei giochi "d'antan" che noti mancare in quelli moderni, diciamo dal 2010 ad oggi?

I giochi dei tempi che furono hanno cuore, immaginazione, fantasia, il loro essere diretti, ma soprattutto divertenti, elemento che titoli da milioni di dollari, con una sceneggiatura degna di hollywood, mai riusciranno a ricreare.

GB: Come scegli i giochi di cui parli? Prediligi una caratteristica in particolare o assegni a tutti pari dignità di trattamento?





Non ho preferenze assolute, anche se ho i miei prediletti ahahahahah. Diciamo che mi piace spaziare un po' tra tutte le macchine e generi senza distinzione alcuna, dall'età della pietra fino alla PS2 (giusto per darmi un limite), ma a volte ho anche oltrepassato il limite.

GB: Il rinato sviluppo di giochi su sistemi d'altri tempi: cosa ne pensi e chi è il tuo programmatore preferito, se ce n'è uno.

É bellissimo vedere quanto è pregno ed agitato il sottobosco degli sviluppatori indie. E' una cosa che valorizza ancor di più le nostre tanto amate macchine, e che consente di sfruttarle molto meglio di trent'anni fa. Provo sempre i nuovi titoli che vengono proposti per tutte le macchine possibili ed immaginabili, è una cosa che adoro davvero, specie quando vengono proposti remake di grandi classici che possono essere comparati con le versioni sviluppate all'epoca (leggasi GnG e Commando di Nostalgia per c64 e tanti altri). Uno sviluppatore? Dai non fatemi fare nomi... va beh, uno lo dico perché adoro il suo lavoro e l'ho anche intervistato su Twitch: Antonio Savona.

GB: Se potessi creare un videogioco, come ti piacerebbe che fosse?

In una parola: D I V E R T E N T E !

GB: Piani per il futuro: dove andrà SalaGiochi1980?

Bella domanda, ma purtroppo non ho la sfera di cristallo. Quello che posso dire é che, nelle mie possibilità, cercherò di portare il tutto avanti con la stessa passione, costanza e divertimento di ora.

Ciao Alex, qui Mic the Biker o come mi chiamava qualcuno "Haggar". Ricordo perfettamente la nascita di sala giochi 80, praticamente ho iniziato a pubblicare la mia passione online proprio grazie a voi. Ti volevo fare qualche domanda.

MtB: Come è cambiata SalaGiochi1980 negli anni?

Ciao Michele, è sempre un piacere fare due chiacchiere con te. Nel tempo SG1980 ha subito molti cambiamenti, ma il leitmotiv è sempre rimasto lo stesso: al primo posto c'è la passione per questo fantastico mondo, tutto il resto è in secondo piano. Ora si dà molto più spazio alle live su Twitch e ai video sul tubo, ma è un'evoluzione naturale,

per rimanere al passo con i tempi.

MtB: Raccontaci della bellissima iniziativa del video di auguri per il primo anno di SalaGiochi1980.

Ahahahahahah, avevo completamente rimosso la cosa. E infatti non ricordo molto, ma sono anche passati 7 anni, l'annite avanza, dai concedimelo.

MtB: Il primo anno fu un turbine di nomi che grazie a SalaGiochi1980 hanno stretto amicizie e creato collaborazioni. Tiz, Clax, il sottoscritto, Edi, Padova e il mitico lightning DJ. Come ricordi quel periodo?

E' stato un periodo bellissimo, eravamo davvero una squadra e sono nate molte belle amicizie che continuano tutt'oggi. Si aspettava la sera di tornare a casa per mettersi a chattare tra noi per scambiarsi pareri, dritte e quant'altro. E non si parlava solo di retrogaming! Lo stato di grazia assoluto per SalaGiochi1980.

MtB: Ultima domanda: parlati delle legendarie classifiche settimanali e del fantastico caos che riuscivano a creare.

La famosissima Top Fans che veniva pubblicata tutti i sabato pomeriggio alle 16. Una vera e propria lotta per ottenere il primato poiché l'app premiava le persone che interagivano di più con la pagina, commenti, like, condivisioni e altro. Ricordo le sfide all'ultimo commento tra di noi per risalire la classifica e vedere il proprio nome almeno nella top ten. Che bei tempi! Grazie per avermeli ricordati! Ciao Mic, sempre al top (fans) ahahahahahah!

Con il ricordo della classifica Top Fans, la nostra intervista si chiude. Ringraziamo Alex per averci permesso di sbirciare dietro le quinte di Salagiochi1980 e gli esprimiamo il nostro apprezzamento per gli sforzi compiuti nel tenere in vita la passione per la retroinformatica che ci accomuna, e gli auguriamo di cogliere successi sempre più ampi ed importanti. Per chi volesse saperne di più, nel box dedicato ai link pubblichiamo gli indirizzi della pagina Facebook, Twitch e Youtube.

Links utili:

- <https://www.facebook.com/salagiochi1980/>
- <https://www.twitch.tv/salagiochi1980>
- <https://www.youtube.com/user/salagiochi1980>





Ricordando Jerry Lawson

Uno dei grandi padri fondatori dei videogiochi moderni

di Takahiro Yoshioka e Carlo Nithaiah Del Mar Pirazzini

Gerald Anderson Jerry Lawson è stato un ingegnere americano noto per il suo notevole lavoro nella progettazione del Fairchild Channel F, una console sviluppata nel 1976 e che all'epoca fu una vera rivoluzione.

È anche considerato l'inventore delle cartucce di gioco e uno dei fondatori dei videogiochi moderni.

Nel 1972, poco dopo l'uscita del leggendario Pong, Jerry Lawson produsse e sviluppò uno dei primi giochi arcade, Demolition Derby. Questa abilità gli è poi valsa una promozione all'interno di Fairchild Semiconductor (una divisione di Fairchild Camera and Instruments) nel ramo dei videogiochi dell'azienda che, grazie a lui, sarebbe diventato uno dei primi produttori di console.

Jerry Lawson ha poi ricoperto la carica di capo ingegnere hardware, direttore dell'ingegneria e del marketing per la divisione videogiochi, lui che, adolescente radioamatore autodidatta, guadagnava soldi riparando i televisori dei suoi vicini.

In un momento in cui le console domestiche non potevano contenere più giochi di quelli incorporati (il Magnavox Odyssey aveva cartucce, ma non contenevano unità di memoria), Fairchild immaginava cartucce di gioco intercambiabili che potevano essere caricate nelle macchine senza danneggiarle.

Da questa idea nascerà nel novembre 1976 la Fairchild Channel F, semplicemente la prima console a cartuccia ROM programmabile, nonché la prima console ad utilizzare

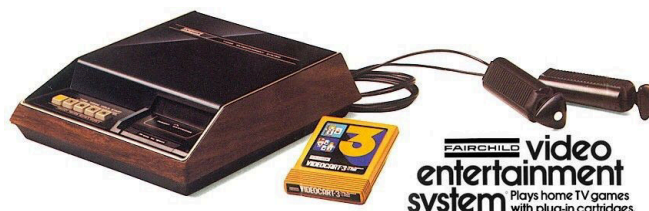


Fig. 2 - La console Farichild

un microprocessore, per la quale Jerry Lawson progettò anche il prototipo di joystick.

È abbastanza chiaro oggi che l'idea originale per la cartuccia da videogiochi sia venuta da due uomini, Wallace Kirschner e Lawrence Haskel, che lavoravano per Alpex Computer Corporation su licenza di Fairchild. Fu allora che un team composto da Ron Smith, Nick Talesfore e Jerry Lawson perfezionò la tecnologia e la trasformò in un prodotto commerciale casalingo. Pertanto, il merito del primo round deve essere tecnicamente condiviso tra questi cinque uomini, e non solo da Lawson.

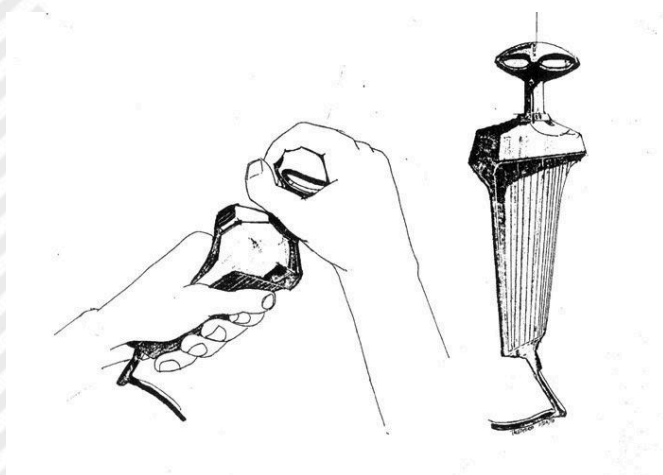


Fig. 1 - Il joystick

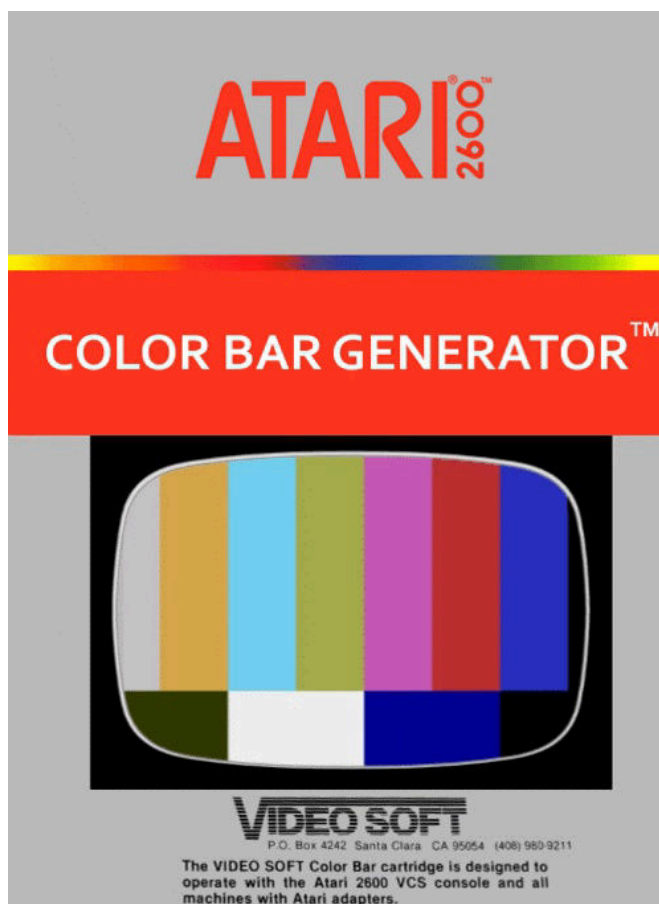


Fig. 3 - La copertina di Color Bar Generator di Video Soft





Fig. 4 - Il Grande Jerry Lawson in una foto recente

Sebbene questo sistema non abbia mai goduto delle vette della popolarità come Atari, Nintendo o Sega, è stato un passo fondamentale in un intero settore perché ha offerto per la prima volta circuiti a cartuccia sufficientemente forti da resistere a shock fisici e a migliaia di inserimenti senza danni. Il timore principale era quello di provocare scariche elettrostatiche che potessero facilmente friggere un chip semiconduttore.

Molto rapidamente, Jerry Lawson divenne membro dell'Homebrew Computer Club, una sorta di lobby di innovatori e club di dilettanti da cui sarebbero poi emerse le più grandi leggende dell'industria informatica, tra cui i fondatori di Apple Steve Jobs e Steve Wozniak. Per la cronaca, Jerry, visibilmente non venne impressionato da Wozniak, gli rifiutò un lavoro alla Fairchild...

Ha anche fondato e gestito VideoSoft, una società di sviluppo di videogiochi per Atari 2600 all'inizio degli anni '80, quando la console è diventata leader di mercato. Sfortunatamente, l'azienda ne ha rilasciato solo uno, uno strumento tecnico chiamato Color Bar Generator.

Illustre "pioniere del settore" dall'International Game Developers Association nel marzo 2011, Jerry Lawson morì un mese dopo a Santa Clara, in California, all'età di 70 anni.

La sua brillante carriera di innovatore lo posiziona come un simbolo per tutti gli afroamericani e come uno dei padri moderni del videogiochi.





Un bit di rarità

(rovistando qua e là)



Warren Davis e l'invenzione di Q*bert

di Alberto Apostolo

Molti videogiocatori amano conoscere gli autori e i retroscena legati alla realizzazione dei loro titoli preferiti.

Da poche settimane, la casa editrice americana Santa Monica Press LLC ha pubblicato "Creating Q*bert and Other Classic Video Arcade Games" scritto da Warren Davis (figg.1 e 2).

Per i pochissimi che non lo sanno, Warren Davis è un game-designer e programmatore, sviluppatore del gioco "Q*bert" (fig.3).

Apparso nel 1982, "Q*bert" divenne uno più famosi arcade game di tutti i tempi entrando a pieno diritto nella cultura popolare con gadget di ogni tipo.

Warren Davis ha anche sviluppato "Us Vs. Them" (un Laserdisc game) e ha aperto nuovi orizzonti nella progettazione dei giochi sviluppando la tecnologia di digitalizzazione che è stata usata in "Mortal Kombat", "Terminator 2 the Arcade Game", "Revolution X", "NBA Jam", "Joust 2", e "Narc". In particolare, per Revolution X, progettò una specie di ambiente pseudo-3D su un hardware-2D.

Con uno stile brillante e amichevole, Warren Davis ci racconta la sua storia ricca di aneddoti interessanti. Si comincia dal periodo scolastico e di come ha imparato a programmare sul Monrobot XI, uno strano computer prodotto dalla Monroe Business Machines.

Nel 1981 sarà assunto presso la Gottlieb (dove svilupperà "Q*bert") e qualche anno dopo lavorerà alla Williams.

La sua carriera professionale si è

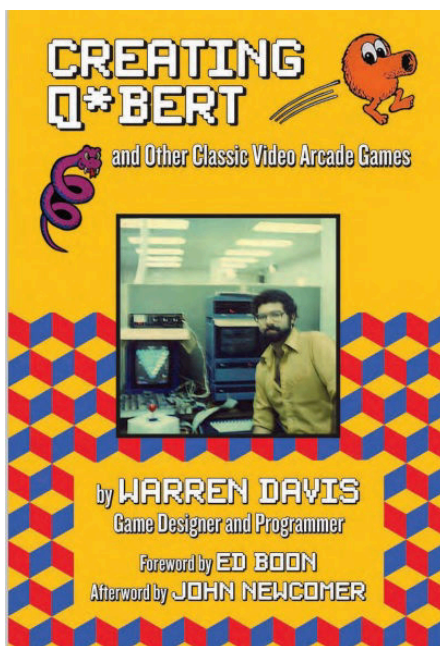


Fig.1 - La copertina del libro

svolta a contatto con i più talentuosi professionisti della storia dei videogiochi: Eugene Jarvis, Tim Skelly, Ed Boon, Jeff Lee, Dave Thiel, John Newcomer, George Petro, Jack Haegar, Dennis Nordman e tanti altri.

Credo di avere detto abbastanza e non voglio togliere il piacere della scoperta, sfogliando le pagine del libro.

Al momento è in vendita solo l'edizione in inglese. Non è certo se sarà stampata anche una edizione in italiano.

Francesco Fiorentini: Avevo promesso ad Alberto di aggiungere il mio commento a quanto già scritto da lui riguardo a questa pubblicazione. Alberto, come di suo consueto, in poche righe ha brillantemente descritto il libro ed il suo contenuto. C'è quindi poco altro da aggiungere al riguardo.



Fig.2 - Warren Davis

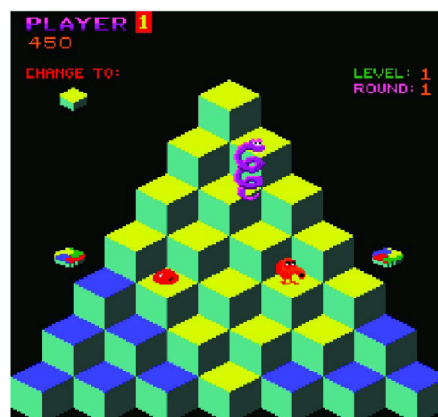


Fig.3 - Il gioco Q*bert

Vorrei però confermare che il libro è scritto molto bene e la lettura scorre fluida, per cui, nonostante la sola versione inglese, mi sento di consigliarne la lettura a tutti, visto che il libro non fa uso di periodi e vocaboli complessi.





A volte ritornano...

di Carlo Nithaiah Del Mar Pirazzini

Titolo: **Mortal Kombat Arcade Edition**

Sviluppatore: **Master Linkuei**

Piattaforma: **Sega Megadrive**

Sito Web: https://drive.google.com/file/d/1TiBqHKLw0a3cl3f5Al4umqNxPpq7E-4_/view?usp=sharing

Mortal Kombat fu una rivoluzione! Travolse noi videogiocatori con una raffica di grafica mai vista prima (ad eccezione di Pit Fighter), giocabilità semplice e ... tantissima ultraviolenza e gore.

Inutile dirvi che la corsa alle conversioni arcade fu devastante. Tutti i sistemi del tempo avevano una conversione più o meno decente del gioco di Midway.

Le versioni che più spiccarono al tempo furono quelle per Amiga, per MS DOS e quelle per le console a 16 bit.

Mentre la versione Super Nintendo subiva una pesante censura dalla stessa Nintendo (risolvibile con un trucco), quella Megadrive, se pur inferiore tecnicamente, era un gran bel gioco.

Dopo moltissimo tempo lo sviluppatore Master Linkuei, famoso per il suo modo di "aggiustare" le versioni arcade per console di molti titoli, ha deciso di rilasciare questo hack che rimette in ordine il gioco portandolo al arcade



perfect, o quasi.

Cosa è stato aggiunto? Abbiamo una pulizia grafica migliore e riedizione della colorazione, le animazioni sono più fluide grazie ad un codice che sfrutta meglio la macchina da gioco. Gli effetti sonori della versione sala giochi sono stati aggiunti totalmente e sono stati abilitati tutti i tips e le modalità segrete non presenti nella versione originale per Sega Megadrive.

Inutile dire che il gioco rimane sempre incredibile da giocare anche dopo tutto questo tempo.

Questa hack rom è liberamente scaricabile, funziona alla grande in emulazione e su real hardware.

Consigliatissimo.





R-TYPE - Shoot'em-up anno zero

di Mic the Biker Novarina

La vedete lì parcheggiata? Proprio qui di fianco. La mitica DeLorean di Ritorno al Futuro è pronta a portarci al termine degli anni Ottanta, precisamente nel 1987. In quell'anno R-Type arrivò in sala giochi e lo sparatutto a scorrimento orizzontale non fu più lo stesso. Una vera e propria rivoluzione nella storia degli shoot'em up. In effetti questo mostro di gioco segue un'infallibile regola del mercato: ti riesce una volta nella vita. Partiamo col nostro viaggio amici, con le nostre 200 lire in tasca.

Entrando in sala giochi e vedendo il cabinato in questione non può non venire in mente quel genio malato di H. R. Giger, alias il creatore di Alien e maggiore esponente dello stile biomech organico. Ma cosa diavolo sarà mai questo gioco? Oggetto misterioso marchiato Iren, è riuscito ad arrivare con il suo fascino intatto fino ai giorni nostri, così come la sua immortale astronave da battaglia R-9 che ha fatto scuola nel design del genere. Uno shoot em up rigorosamente orizzontale con arsenale potenziabile con quattro diversi tipi di arma, grafica fuori scala nel vero senso della parola e sonoro esaltante.

Niente fino a quel giorno si era spinto tanto oltre: c'era la fila dietro al cabinato anche senza nessuno che ci giocasse! Ma a parte grafica mai vista e sonoro da manicomio c'era realmente altro? Assolutamente sì. Era dall'alba dei primi sparatutto che non fu introdotta alcuna novità di rilievo. Qui in primis abbiamo il Force Device, la famosa sfera diventata d'obbligo da lì in avanti che raddoppia la potenza di fuoco. Può anche essere lanciata contro gli avversari o controllata e attaccata davanti o

dietro l'R-9, in modo da poter sparare in entrambe le direzioni! Come non bastasse il Force Device assorbe i colpi meno potenti degli avversari, divenendo uno scudo vero e proprio. Idea assolutamente geniale, accompagnata, naturalmente, da una dinamica che ne consente lo sfruttamento in maniera intelligente, diminuendo quella spasmodica necessità di sparare sempre più velocemente. E proprio grazie a questo mix che si eleva il divertimento. Perché R-Type è incredibilmente divertente! A differenza dei suoi predecessori, non basta memorizzare i percorsi dei nemici: qui si deve fare pratica per impiegare le risorse a propria disposizione in maniera appropriata al livello o alla serie di nemici in arrivo. Già si fosse fermato qui sarebbe stato sufficiente per annichilire i suoi concorrenti ma no...

Arriva il Beam!

Come se non bastasse R-Type aggiunge anche un altro elemento: il Beam! Si tiene premuto il pulsante di fuoco e si vedrà caricare una barra di energia. Una volta al massimo, l'R-9 rilascerà un devastante raggio in grado di risolvere le situazioni al limite dell'umano. Pensate a quanto sia stato rivoluzionario tutto ciò: si doveva pensare nel giro di pochi secondi il vero beneficio dovuto al tempo perso per caricare l'arma ed all'effettivo danno causato agli incredibili nemici. Ecco, mi volevo soffermare proprio qui. Mai prima di allora dei nemici potevano essere fuoriusciti da un incubo: il primo mostro enorme, il cuore biomeccanico grosso come tutto lo schermo, sono cose che abbiamo sognato di notte. Per la prima volta i nemici, i mostri, sono una colonna portante di un gioco, fondamentale anche per l'evoluzione del filone. Maligne creazioni biomeccaniche, agghiaccianti, ancestrali, originali e molto, molto grosse. Spaventosamente grosse. I guardiani finali sfoggiano senza pietà dimensioni full screen annichilenti! L'R-9, che fino a qualche istante prima sembrava immensa, diventa in men che non si dica un granello di sabbia di fronte alla maestosità degli xenomorfi, giusto per fare un altro tributo a Giger.

I boss hanno un punto debole

Giganti dai piedi di argilla, perché anche qui la dinamica è diversa: vanno giù con pochi colpi, ma vanno sparati in





un minuscolo punto debole. Una gara di nervi, perché i colossi in questione sparano. Quanto ci siamo accaniti, quanti gettoni ci abbiamo buttato in questo allucinante coin op, non fosse altro per vedere come fosse la grafica più avanti!

Certo che, come tutti gli sparatutto, non è un gioco perfetto, il genere stesso non lo può essere. Ma lo shoot em up è l'essenza della sala giochi! La difficoltà del gioco è calibrata in modo particolare. Partiamo dal presupposto che, quando si diventa abili, si può finirlo dopo poche partite senza morire mai. I problemi si hanno quando si viene uccisi e si riparte da un punto precedente troppo arretrato e, soprattutto, senza armamento! In questa condizione è impossibile uscire vincitori. Ma noi non ci piegavamo: dentro un altro gettone, si deve superare quel dannato livello.

Tutti noi abbiamo giocato almeno una volta a questo gioco, tutti noi abbiamo a nostro modo partecipato a renderlo un capolavoro immortale ed uno bivio nel suo genere. Dopo di lui nulla fu più uguale. Nel 1987 il suo

impatto fu come il monolite di 2001 odissea nello spazio: elevò il genere. R-Type fece estinguere il vecchio concetto di shoot'em up, lo cambiò, lo fece evolvere e lo rese migliore. Ricordo ancora la frenesia di giocarci, di fare a rotazione con gli amici per passare i livelli. E ricordo con che ansia noi giovani utilizzatori aspettavamo notizie su eventuali conversioni per i nostri home computer o console.

E arrivò Katakis

Ogni numero di Zzap! veniva sondato riga per riga alla ricerca di conferme che, visto il successo galattico del gioco, non tardarono ad arrivare. Iniziarono a trapelare notizie di una conversione per il Commodore 64, mio home computer all'epoca. La Activision aveva annunciato in pompa magna la realizzazione del gioco. Ma mentre noi facevamo il conto alla rovescia su un numero di Zzap! appare la recensione di un gioco della Rainbow Arts chiamata Katakis.

Fulmini e saette ma questo è R-Type col nome diverso! Ne nacque una disputa legale e il titolo fu ritirato dal mercato. Nel frattempo noi tutti avevamo già messo le



**Rtype versione Commodore 64**

mani su qualche copia pirata da tempo, ed era veramente un gran bel gioco. La Activision vide la bravura del team Factor 5, che sviluppò il gioco, e li assunse per la conversione Amiga uscita nel 1989. E così poco dopo la Rainbow Arts poté far uscire di nuovo Katakis sotto il nome di Denaris. Le telenovole erano di moda all'epoca, anche nel nostro settore!

Conversioni domestiche

Parliamo un po' delle migliori versioni casalinghe di R-type. Per il buon **Commodore 64** la conversione, considerata la macchina in questione, fu soddisfacente ma forse, vista la portata del titolo, un po' sopravvalutata. Iniziamo dal porting vero e proprio, tralasciando la grafica: alcuni nemici non seguono perfettamente le traiettorie di attacco che erano presenti nel coin-op. A mio avviso questo aspetto taglia decisamente le gambe alla conversione perché gran parte della giocabilità si basa sui tipi di attacco portati dai nemici. Togliendo questo fattore si riduce il tutto ad un banale sparatutto. I controlli peccano della mancanza del secondo pulsante e il dover premere la barra spaziatrice per richiamare il force non è comodo ma di meglio non si poteva fare. Il medesimo, quando è staccato dall'R-9 tende a fare traiettorie inutili.

Finiti i difetti andiamo a vederne i pregi. La grafica è generalmente di buona fattura ed è molto rassomigliante alla versione originale, compreso un basilare parallasse nel fondale. Il sonoro dal canto suo è stratosferico: il buon SID è una garanzia e spremuto a dovere tira fuori colonne sonore perfette. La conversione riuscì così a mantenere quella atmosfera di biomech che ha reso celebre l'originale.

La conversione che colpì di più il sottoscritto fu quella per il glorioso **Zx Spectrum**. Pur avendo i suoi conosciuti limiti hardware rispetto ai concorrenti, i programmatori

**Rtype versione ZX Spectrum**

riuscirono a rendere il comparto grafico splendido. La buona velocità del processore permetteva una giocabilità fluida e precisa. La resa finale fu invero molto colorata, con l'effetto blocchi limitato e un dettaglio stupendo. Un mio Carissimo amico aveva lo Spectrum e come da tradizione ci si attaccava tra c64isti e spectrumisti, ma questa volta mi levai il cappello.

Restando sul pianeta home computer passiamo alla versione **Amiga**, il 16 bit di casa Commodore. Come accennato prima, la conversione è datata 1989 per merito dei Factor 5. Questi programmatori teutonici fecero questo primo buon lavoro che li mise in luce, anche se la conversione ha grosse lacune. R-Type in versione Amiga non è un prodotto impeccabile: viene a perdersi la fedeltà della conversione pur a fronte di un ottimo lavoro di programmazione. I difetti marchiani qui sono le dimenticanze rispetto all'originale, e ci può stare, ma anche in confronto ad altre trasposizioni per sistemi molto meno performanti, e qui non è perdonabile. Mancanze perlopiù grafiche che lasceranno l'amaro in bocca perché, viste le caratteristiche della macchina, si poteva puntare ad una conversione quasi perfetta. Nella versione Amiga di R-Type manca

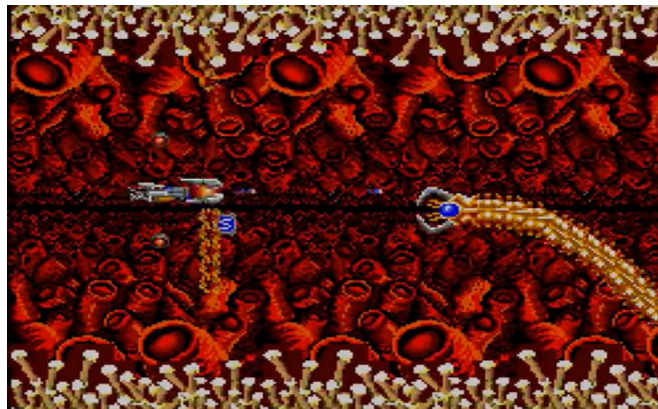
**Rtype versione Amiga**



Rtype versione PC Engine

qualsiasi tipo di parallasse presente nella versione originale. Una mancanza incredibile visto che Amiga riusciva a gestire il parallasse come pochi altri, basti vedere *Shadow of the Beast*. L'effetto grafico del caricamento del beam è sparito. Tanti fondali sono stati rimossi, tipo quello del terzo livello che circonda l'astronave nemica. Quest'ultima presenta meno parti da distruggere, è stata resa troppo facile. È stato tutto il game play ad essere stato semplificato, troppo a dire il vero. Come scritto all'inizio, il coin op non è certo impossibile da domare, ma richiede buona esperienza da videogioctore navigato e minuziosa padronanza dei controlli. La versione Amiga, anche paragonata al concorrente 16 bit chiamato **Atari ST** (di cui potete trovare una minuziosa recensione sul numero 5 di *Retromagazine*), ne dimezza drasticamente il contenuto. Faccio un altro esempio: il verme biomech del secondo livello è notevolmente più corto anche della versione Atari ST. Forse sono stati peccati di inesperienza del team di programmazione e una relativa gioventù della macchina: chiunque può notare le differenze dalla controparte originale.

Ho lasciato volutamente per ultime le due conversioni migliori. Perché sono veramente incredibili. La prima è quella per **PC Engine**. Fu un connubio storico questo, un binomio certamente casuale: grazie a questa conversione il PC Engine venne conosciuto dal mondo intero. La macchina in questione spopolava in Giappone ma era quasi un oggetto misterioso nel resto del mondo. Furono riprodotti i primi quattro livelli del coin op ma la cura e la fedeltà all'originale furono assolute. Fu una scommessa vinta: uno dei più popolari arcade del periodo fu trasportato su un'apparecchiatura domestica con quasi nessun tipo di mancanza o perdita grafica e sonora. Ci fu un periodo che quando dicevi R-type dicevi PC Engine e viceversa. Purtroppo in Europa arrivarono pochissime macchine con



Rtype versione Master System

nome Turbografex e restò per quasi tutti un sogno, reso possibile solo in tempi recenti con la messa in commercio della versione Mini, a mio avviso l'unica vera replica delle vecchie console che ha veramente senso.

E per ultimo guardiamo cosa riuscì a combinare Sega per la sua console 8 bit. Diciamolo chiaramente, per il **Master System** questa trasposizione ha dell'incredibile: anche se gli sprite presenti su schermo sono leggermente in minor numero tutti gli scenari si avvicinano di molto alla versione originale! La grafica è così definita e colorata che guardando il gioco sembra di vedere uno dei primi giochi per Mega Drive. Il dazio da pagare per un comparto grafico simile è qualche sfarfallio e alcuni rallentamenti. Il sonoro lascia basiti ancora di più del reparto grafico per somiglianza al coin op, anche perché l'otto bit di casa Sega non era particolarmente fornito di chip dedicati. Dal punto di vista della giocabilità siamo alla perfezione e la implementa con una professionalità tale da non far rimpiangere l'originale da bar. Come abbiamo visto prima il flusso dei nemici era uno dei punti forti del coin op e qui sono praticamente uguali, sia la maggior parte degli schemi di attacco che la loro posizione sullo scenario. Glielo vogliamo trovare un neo? Il ritmo di gioco è leggermente più lento di quello originale e c'è qualche rallentamento. Ma quelli di Sega hanno voluto fare di più: la presenza di un intero livello con relativo boss segreto! Vi svelo come si accede? No dai, dateci una rigiocata, non ve ne pentirete.

È ora di tornare bambini, forza! Blast off and strike the evil Bydo Empire!





The Age of Heroes

di Mic the Biker Novarina

La DeLorean oggi non serve, perché andremo a fare la conoscenza di un gioco nuovo, o relativamente tale, per il nostro amato biscottone. E' ormai assodato che il mondo del retrogaming non è più solo una stanza polverosa per vecchi nostalgici. Quello che inizialmente, anni fa, era visto con simpatia come un movimento di anziani che rievocavano vecchie glorie, oggi è esploso in qualcosa di nuovo. Un'onda con la forza di uno Tsunami composta da nuovi e fantastici giochi, la quale viene cavalcata dalle poderose vecchie piattaforme 8 e 16 bit, oggi più in forma che mai. Ecco, The Age of Heroes fa parte di questo nuovo che avanza!



Mentre il nuovo claudica e zoppica, complice un periodo storico non proprio luminoso e una crisi della componentistica elettronica mai vista fin ad ora, il vecchio si tira a nuovo e sfoggia un fisico da dieci e lode. I nuovi sistemi di intrattenimento sono quasi fermi al palo, mentre le vecchie glorie guadagnano sempre più spazio nel cuore degli appassionati, anche under 40. Certo, io sono di parte, sono cresciuto con questi sistemi e li adoro ma obiettivamente hanno parecchi pregi. Prendiamo ad esempio il buon Commodore 64: mai ricevuto un aggiornamento di sistema in 40 anni. I suoi giochi? Li pigli, li carichi e ciao. Puoi farlo sulle macchine originali

degli anni 80, lo puoi fare sulle riedizioni in chiave mini o maxi o li puoi emulare quasi ovunque. E il discorso vale per le restanti vecchie glorie del passato.

Contrapposto a questo, trovo snervante la gestione delle console next gen, da Play 4 e Xbox one in avanti. Le accendi e c'è già un aggiornamento di qualche giga che ti aspetta. Ok, ma cosa ci sarà da aggiornare continuamente? Finito l'aggiornamento inserisco il gioco, perché ho preso da Game Stop l'edizione fisica che mi è costata meno di quella digitale. Magia, neanche il tempo di vedere un'immagine e mi appare una schermata che mi avvisa che ci sono 30 giga di aggiornamento per far partire il gioco. Ma sul disco cosa ci hanno messo allora? Ed è un continuo, tra DLC, altri aggiornamenti, upgrade. Poi il gioco dopo una settimana mi ha rotto e lo butto in un angolo.

Ecco, forse ho capito...

Ecco che inizio ad aver chiara la situazione! Se ho mezz'oretta e mi voglio distrarre un attimo dalla valanga di casini che giornalmente la vita ci rovescia addosso, il retrogaming fa esattamente quello per cui era nato più di 40 anni fa: divertire! Ecco che col tempo tutti se ne sono accorti e lo sviluppo di nuovi giochi ora non è più un fatto raro ma una realtà periodica molto solida. Oggi vediamo uscire giochi per le nostri vecchi arnesi elettronici che fossero usciti ai loro tempi ci avrebbero fatto divertire molto, e in alcuni casi avremmo urlato al miracolo.

Ed eccoci a vedere questo divertente Hack n Slash che trasuda RASTAN da tutti i pori. Psytronik è una garanzia, ci ha abituati da sempre a giochi buoni e dalla dinamica notevole. Non fa eccezione questo The Age of Heroes, uscito già da un paio di anni ma, essendo parte di questo "ritorno al futuro", forse non molti conosceranno. Il team che si è occupato del gioco è di prim'ordine: programmato da Achim Volkers, alla grafica abbiamo Trevor 'Smila'





Storey, mentre il sonoro è una garanzia grazie a Saul Cross. La storia ci racconta che uno spettro malvagio sta inviando la sua legione di dannati attraverso le lande ghiacciate, le montagne e i templi sacri. Saremo noi, protagonisti impavidi col Joystick in mano, a dover porre rimedio a questo scempio, viaggiando e lottando per eliminare lo spettro e le sue legioni. Del resto siamo noi discendenti dei guerrieri della luce, possiamo esimerci dal combattere?



Il gioco

Dopo una gran bella schermata di caricamento, e una di selezione tra un guerriero maschio o femmina, Iniziamo il nostro viaggio. Eccoci alle prese con un arcade a scorrimento laterale, armati solo di una spada e di un incantesimo. Come nella miglior tradizione del genere, i nemici attaccheranno a ondate da entrambi i lati, cercando di renderci difficile la vita. La grafica è di livello, il nero è usato sapientemente per creare contrasti elevati e fondali appaganti. Gli sprites sono ben animati e il sonoro è una garanzia. Il feeling è bello, praticamente è come tornare indietro nel tempo. Diciamolo subito, ovviamente la dinamica di gioco è quella, non può essere altrimenti. Da apprezzare la volontà di inserire qualche variante, come ad esempio nel secondo livello: qui troveremo un boss da eliminare. Molto bella, sia per l'atmosfera che per dare più profondità al tutto, l'idea della mappa cartacea che ci evidenzia i livelli bloccati o meno.

La mia gioia è sempre al top quando mi trovo davanti a gemme old school come queste. Ciò che mi ha sorpreso in positivo è la difficoltà iniziale a livelli umani: si arriva fino al primo boss con relativa tranquillità. Certo, la dinamica è sempre la medesima: si procede, si salta, si usa la corda per salire e scendere e si menano fendenti. State pensando "Come in Rastan!", ed avete ragione. Credo che gli sviluppatori abbiano preso proprio il famoso arcade come esempio da seguire, rendendo più semplice l'approccio iniziale. Questo a causa di alcuni feedback dei giocatori in fase di beta testing. Così il team di sviluppo ha deciso di sbloccare il livello più difficile solo una volta completato il gioco.

Un aspetto che ho gradito è quella che nel gioco si può salire di livello. Come ogni buon gioco di ruolo che si rispetti, uccidendo nemici si accumulano punti, i quali ci faranno avanzare e divenire più forti. L'effetto grafico che accompagna questo aumento è grandioso, ed è forse la cosa più bella di questo aspetto di gioco. Quando si sale di livello, recuperiamo tutta l'energia, e ne viene aggiunta un po' di più. Mi ha deluso il modo in cui si accumulano i punti per avanzare: solo le uccisioni in mischia contro nemici danno punti. Se usiamo la magia non succede nulla, tanto meno uccidere un boss. Ci sono alcuni bug nel gioco, o almeno io li ho reputati tali. Può succedere di atterrare leggermente troppo in alto per colpire i nemici, non ci riuscivo nemmeno abbassandomi. Alcune volte i cattivi mi hanno attraversato senza farmi nulla, il che non mi ha dato fastidio! In definitiva niente di così grave da inficiarne la giocabilità, del resto anche un mostro sacro come Rastan per C64 aveva qualche problema. Il gioco è disponibile in diversi formati: si va dalla cassetta standard, ad una speciale edizione su nastro a conchiglia (limitata a 50 copie e credo già finite da un pezzo). Abbiamo poi due versioni su disco, una economica e una premium. Il top resta il cofanetto Collector's Edition con il gioco su disco, CD della colonna sonora, poster, portachiavi, badge e adesivi! Vi posto i link per i dettagli su come ordinare le versioni fisiche, nonché il download da csdb.

<https://psytronik.bigcartel.com/product/the-age-of-heroes-c64>

<https://csdb.dk/release/?id=176388>

GIUDIZIO FINALE

GIOCABILITA': 90% Non si può non pensare a Rastan giocandoci, ma per fortuna il gioco non si limita a questo. L'azione non manca ed è ben calibrata: menare fendenti è sempre un gran piacere.

LONGEVITA': 75% Come la maggior parte di questo genere di giochi, una volta terminato lo si ricaricherà di rado. Peccato, visto che solo finendolo si attiva la difficoltà di gioco pensata inizialmente dai programmatori, vera sfida da ritentare!





NEW GAME

THE SHADOWS OF SERGOTH

Anno: 2020

Editore: Double Sided Games

Genere: Gioco di ruolo

Piattaforma: Amiga

Sito web: <https://doublesidedgames.com/shop/commodore/commodore-amiga/the-shadows-of-sergoth/>

Un bel Dungeon Crawler vecchio stampo su Amiga? Che meraviglia! Amavo la serie di Eye of The Beholder e questo The Shadow of Sergoth ne è un erede.

Il gioco si sviluppa nel mondo fantasy medioevale di Chrisandia, nella penisola di Marak.

Una zona non molto popolosa dove la popolazione si guadagna da vivere con il commercio e la pesca.

Il suo attuale Re, Orlof VI il Valoroso, è un membro di una lunga dinastia di regnanti illuminati. Durante le guerre contro il malvagio imperatore Sul Rakin, Re Orlof dimostro grande valore e il regno ottenne una vittoria sul male ma tantissime perdite.

Gli eventi nel gioco si svolgono cinque anni dopo la caduta di Sul Rakin, nell'anno 351 dell'era del cristallo.

Si tratta di un classico gioco di ruolo d'esplorazione sotterranea che ricorda i già citati Eye of the Beholder o Dungeon Master, si basa su Microlite20, una semplificazione delle regole dell'Advanced Dungeon & Dragons e nasce basandosi sulla versione del gioco per Amstrad CPC6128 che uscì nel 2018.

Il gioco necessita di un Amiga con almeno 2 Mega di chip Ram anche se personalmente consiglio un Amiga più veloce (Amiga 1200 o accelerato) per attivare gli effetti di movimento 3d e migliorare la fluidità d'azione.



Il gioco ad ogni modo funziona su qualsiasi chipset grafico (OCS, ECS o AGA) e gira configurato anche in emulazione WinUAE e su PiMIGA.

Il personaggio, come nel più classico dei gdr, può aumentare esperienza e quindi crescere di livello fino al





GIUDIZIO FINALE

» Giocabilità 80%

Sistema di gioco classico ma divertente. E' un dungeon crawler e andare a caccia di mostri è divertente. Belli gli incantesimi presenti.

» Longevità 75%

Il gioco si lascia giocare bene ma i dungeon non sono impegnativi, si rischia di finire anche troppo presto l'avventura. Divertente la genesi del party.

ventesimo. Ovviamente sarà possibile configurare il proprio personaggio nel modo più classico e livellare proprio come in AD&D.

Sono presenti 5 razze molto particolari; oltre ad umani, nani ed elfi potremo giocare anche il mezzorco e uomo lucertola. Questo permette un party molto vario e particolare (il lizardman è meraviglioso).

Il sistema di mappatura è fortunatamente automatico e questa cosa ci sarà davvero utile rispetto al mappare come ai vecchi tempi.

Sono presenti 20 dungeon molto grandi da esplorare ed è presente un buon sistema di salvataggio.

E' anche possibile installare il gioco su disco fisso (consigliato).

La scatola contiene tre dischetti da 3"1/2, il manuale in inglese, una bussola vera che potrete utilizzare nei dungeon (folle ma vero, ndN), un D20 da collezione, diversi adesivi e il link al download degli ADF.

Dopo avervi descritto tutto quanto passiamo alla valutazione.

E' un erede degno della serie della SSI o di Dungeon Master? Non del tutto o almeno è un bellissimo gioco ma non arriva alla perfezione della narrazione dei giochi citati.

L'avventura è giocabile e lineare e presenta un buon grado di sfida ma i dungeon sono "semplici".

Graficamente è un gioiellino e devo ammettere che si sono dati un gran da fare per renderlo gradevole.

Vale la stessa cosa per il sonoro, d'atmosfera e funzionale.

Però.... Però manca quel quid in più per renderlo un gioco incredibile, un gioiello assoluto.

E' un bel gioco che merita un voto di tutto rispetto, ma sul gradino la saga di EOB resta stabile.

di Carlo N. Del Mar Pirazzini





NEW GAME

PUZZLE BOBBLE

(COMMODORE 64)

Anno: 2022

Sviluppatore: Bas Scheijde (aka Acied)

Genere: Puzzle Game

Piattaforma: Commodore 64

Sito web: <https://csdb.dk/release/?id=213965>

Sono in esilio causa COVID da zio Nithaiah che in questi giorni è esaltatissimo! È uscito Puzzle Bobble per Commodore 64 e non sta più nella pelle.

Mi passa il Commodore 64, mi spiega come fare a farlo andare e mi dice "Ingrid c'è da scrivere due righe su questo titolo, sei una sua giocatrice da quando eri grande come una dei draghetti... dimmi cosa ne pensi?". Non potevo rifiutare... e ho fatto bene!!!

Puzzle Bobble è un gioco puzzle arcade realizzato da TAITO nel 1994 e poi convertito per alcune console successivamente. Il gioco riporta al successo commerciale i due teneri draghetti del videogioco Bubble Bobble sempre di Taito.

Il giocatore deve usare un cannone per sparare palline colorate causali (bolle) in una serie di palline colorate sparse nello schermo di gioco. Queste palline sparate si attaccano a quelle presenti e, se si riesce a formare un gruppo di tre o più elementi colorati allo stesso modo, queste si dissolvono. Si hanno circa 5 secondi per ogni lancio. Se questo non accade, la palla verrà sparata automaticamente. Lo scopo è ovviamente quello di eliminare tutte le palline colorate nel livello entro un certo limite di tempo e proseguire. Di tanto in tanto il campo di gioco si rimpicciolisce nella direzione più bassa (il soffitto si abbassa). Se



una delle palline in campo raggiunge la parte inferiore, ovvero l'area dove sono presenti il cannone, Bub e Bob avviene il game over.

Velocità e precisione. Essenziali per superare il livello.





Nessuna versione commerciale ufficiale è stata rilasciata per il Commodore 64. Nel 1994 la macchina 8 bit di Commodore era già sulla strada del tramonto.

Il progetto è stato lungo e ambizioso. Riportare su una macchina a 8 bit l'aspetto del gioco nel modo più fedele possibile sembrava impossibile ed invece.

È tutto bellissimo e c'è tutto quello che noi fanatiche di Puzzle Bobble cerchiamo! Funziona tutto benissimo. Mi è piaciuta la grafica, colorata e animata davvero in modo grandioso. C'è persino la schermata iniziale con la scritta Neo Geo. Graficamente un gioiello davvero. Impensabile per una macchina così "anziana e limitata".

Anche il sonoro mi è sembrato appropriato e gradevole, forse leggermente sottotono rispetto all'aspetto grafico, ma perfetto nel contesto.

Il gioco supporta sia il joystick, sia il pad che i paddle. È presente la divertente modalità torneo a due giocatori che si

effettua con il tasto numerico corrispondente:

1. Casuale (livelli casuali)
2. Random Endless (livelli casuali infiniti)
3. Livelli (livelli in ordine di gioco)
4. Livelli infiniti (livelli in ordine ma infiniti).

La versione a due giocatori regala a questo titolo l'eternità!

L'autore nel caricamento iniziale ha inserito un trainer che permette di saltare alcuni livelli.

Che dire di più. È Puzzle Bobble, uno dei titoli più giocabili mai realizzati. Divertente in singolo, mai scontato il livello di difficoltà e praticamente eterno. Scaricatelo e supportate la scena dei nuovi sviluppatori.

Una curiosità. Questa versione utilizza alcune schermate di caricamento dove vengono presentati i progetti futuri di questo incredibile team di sviluppo.

Dimenticavo... non perdetevi l'intervista a Bas Scheijde su questo stesso numero. Io torno a giocare.

di **Ingrid Poggiali**

GIUDIZIO FINALE



» Giocabilità 99%

Semplice, bilanciato e divertente in tutte le sue opzioni di gioco.

» Longevità 95%

32 livelli in singolo sono una bella sfida... ma in doppio e con le diverse opzioni di gioco diventa eterno.





NEW GAME

WIZARD OF THE BOARD

Anno: 2021**Sviluppatore:** Daniel Savage**Genere:** Gdr/Puzzle/Action**Piattaforma:** Nintendo 64**Sito Web:** https://mega.nz/file/h35G0A5b#zOA2MQtH_BwOaU_btPDiceXnudm60APAqvJMKC SYfA

Wizard of the Board è un gioco homebrew per Nintendo 64 realizzato da Daniel Savage e dal suo team e pubblicato il 7 dicembre 2021 per la 64bre Game Jam, una gara di sviluppatori e appassionati del Nintendo 64.

Nel gioco interpreteremo un giovane guerriero di nome Zeff che si vede costretto ad affrontare demoni, mostri e maghi all'interno di una torre e lo farà con l'utilizzo.... Degli scacchi!! Il gioco ha vinto la Game Jam per originalità, aspetto grafico e gameplay innovativo.

Si tratta di uno dei pochi giochi homebrew dotato di una storia di accompagnamento e questo ci stupisce. Spesso nelle competizioni si fa tutto il più lineare possibile senza dedicarsi al "background" del titolo in concorso.

La storia ci viene illustrata tramite diverse schermate introduttive e, come dicevamo, ci narra le imprese del giovane Zeff che sta facendo di tutto per diventare un prode guerriero. L'ultimo passo del suo addestramento è quello di addentrarsi nella Torre e cimentarsi in numerose prove prima di affrontare il Mago del Consiglio. Il gioco è fatto di incontri di tutti i tipi, basti pensare che il primo personaggio che incontreremo nella torre è un topo in cerca di formaggio, piuttosto amichevole.

Spesso durante il gioco avremo dei flashback del protagonista e del suo

allenamento.

I nemici demoniaci che abitano la torre sono parlanti e comunicano al nostro povero protagonista le terribili torture che gli faranno.

Il gameplay del Wizard of the Board non è al primo impatto semplice, ma ha senso e diventa avvincente una volta che ci si prende la mano. I poteri del nostro eroe spostano pezzi di scacchi giganteschi lungo una scacchiera (il pavimento) secondo gli stessi schemi di movimento che farebbero in una normale partita di scacchi da tavolo. L'obiettivo è posizionare tutti i pezzi sugli spazi bersaglio e, nelle fasi successive, sconfiggere anche i nemici presenti sullo schermo.

Potremo farlo selezionando un pezzo guardandolo e premendo il tasto A del Pad, quindi selezioneremo una destinazione target con i pulsanti C o uno spazio vicino con il pulsante Z. Questo può sembrare molto confusionario per via della prospettiva in prima persona, ma con un po' di pratica il gioco diventa divertente e coinvolgente.

Ci sono cinque tipi di demoni presenti contro cui dovremo lottare. I Frogger, grosse rane che saltano da sinistra a destra e viceversa. Fastidiosi ma prevedibili.

Le Blade Traps che si muovono in linea retta fino a raggiungere un ostacolo, dopodiché cambiano direzione. Seguono un percorso fisso fino a





quando non posizioniamo il nostro pezzo, a quel punto cambiano marcia. Ostici.

Gli Smiley Spitters, strani mostri gialli che sparano proiettili, e quindi saremo costretti a muoverci in fretta per evitarli. I proiettili sono lenti ma decisamente letali.

I Blue Demons, enormi demoni che ci inseguiranno senza sosta se entri nel loro campo visivo. Bisogna nascondersi e pianificare. Vanno messi KO con tre colpi.

Demon Queen, fluttua sul tabellone rimanendo lontano dal giocatore ma sparando proiettili. Ostica.

Il sistema di controllo sembra traballante all'inizio, ma una volta compresi i comandi tutto diventa uno spasso.

Il personaggio si muove con il D-pad oppure con la leva del joystick mentre con la pulsantiera C sposteremo il cursore. Frenetico in caso di assalto nemico, ma divertente.

Nonostante sia un genere completamente diverso, Wizard of the Board ci ricorda un altro gioco di Daniel Savage del 2020, Lunar Assault 64 (un bel titolo shooter). Ha livelli ben distinti e dialoghi intermedi che spiegano la storia, l'ambientazione ricorda il mondo lunare già visto nello shooter e persino i nemici somigliano ai mostri spaziali. Chissà forse il creatore del gioco realizzerà una saga collegata. Staremo a vedere.

Unire scacchi, puzzle game e Fps è ambizioso ma decisamente unico. Elaborato e complesso all'inizio per prendere dimestichezza con la meccanica di gioco, ma divertente.

Non è punitivo. Gli scontri con i demoni arriveranno solo verso la metà del nostro viaggio, dopo che avremo compreso perfettamente cosa fare. Un punto a favore. È raro trovare homebrew che ti guidino perfettamente nella meccanica.

È consigliato un controller 64 se si gioca in emulazione per avere a disposizione la corretta mappatura dei tasti. Il gioco è stato rilasciato in versione PAL e NTSC e funziona perfettamente su real hardware tramite Everdrive.

I puzzle non sono complicati da terminare ma offrono una difficoltà bilanciata e, una volta portati in fondo, una bella sensazione di vittoria.

Concludendo.

Wizard of the Board è una bella sensazione di gioco. Sembra di esser tornati nell'epoca d'oro dei titoli N64 soprattutto per lo stile di progettazione. Mi è piaciuta la progressione dei livelli con i brevi filmati di intermezzo (che secondo il creatore sono ispirati a Puyo Puyo), aiutano a rendere il gioco più di una semplice serie di livelli.

Ha una lunghezza decente per un gioco homebrew e sembra decisamente più complesso di alcuni titoli recenti o passati per N64.

Unico appunto lo posso lasciare sul comparto audio non proprio di primissimo ordine, ma sono piccole cose. Nel complesso un ottimo gioco che potrete completare in circa 1 ora di gioco.

Meritatissima la vittoria alla competizione 64BREW Game Jam 2021.

di **Carlo N. Del Mar Pirazzini**

GIUDIZIO FINALE



» Giocabilità 80%

Un mix tra fps, puzzle, scacchi e action game. Livelli ben strutturati.

Ostico il sistema di controllo inizialmente.

» Longevità 90%

Decisamente ben bilanciata la difficoltà del gioco e la sua durata.

N64brew



**NEW GAME**

SILVER VALLEY

Anno: 2017 digitale - 2022

edizione fisica

Sviluppatore: Enriquez Ruiz**Genere:** Action/Metroidvania**Piattaforma:** Sega Master System**Sito:** <https://www.smspower.org/Homebrew/SilverValley-SMS>

La scena homebrew del Sega Master System, la console 8 bit della Sega, è molto viva ed attiva e negli anni ci ha regalato alcuni titoli davvero singolari e accattivanti.

Uno di questi è Silver Valley, un action che definiremo in chiave moderna Metroidvania.

In realtà il titolo presenta una varietà di elementi diversi che non lo catalogano in un genere specifico, ma per comodità e soprattutto per visione ricorda i classici come Castelvania o Metroid.

La storia è molto classica. Liberare il mondo del nostro protagonista dal male. Fine.

Lo stile di sviluppo del gioco è un connubio tra i classici Konami e Capcom del tempo e alcuni titoli tipicamente british che vedevamo spesso sul mondo Amiga.

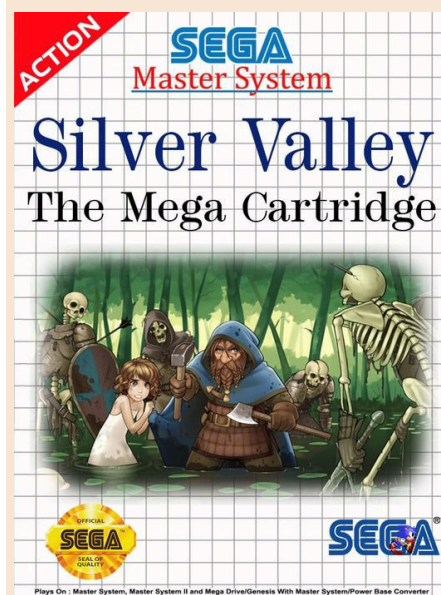
Controlli reattivi che con due tasti ci permettono di saltare, attaccare gli avversari, aggrapparci ad oggetti vari, strisciare e via dicendo.

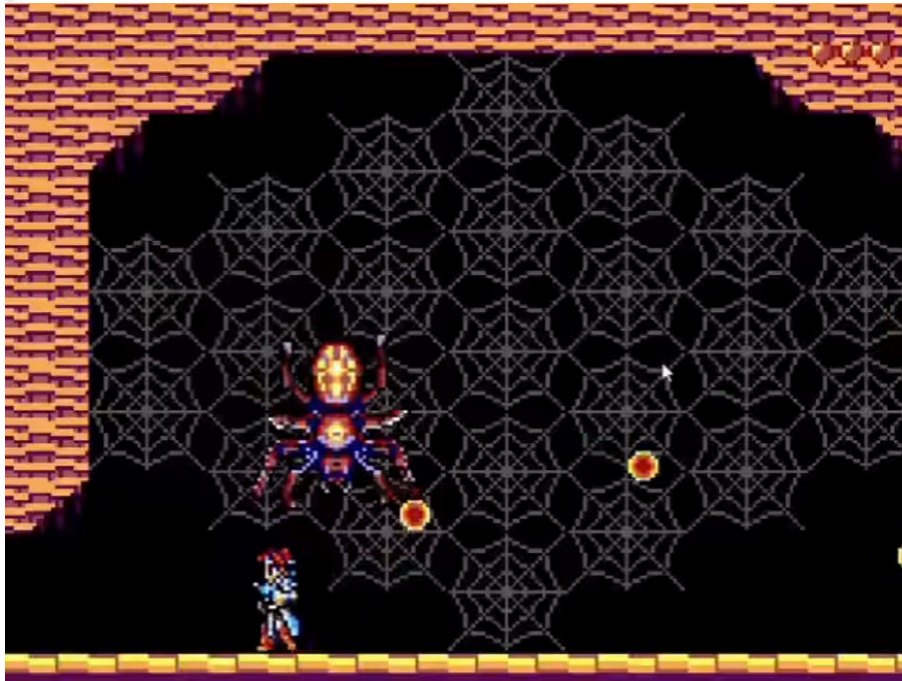
Tecnicamente e visivamente è forse uno dei titoli più belli e vari per la console a 8 bit, almeno della nuova era degli homebrew.

Alcuni livelli sono coloratissimi, come ad esempio quello in stile shoot em up, altri ricchi di animazioni. Lo sforzo degli sviluppatori è stato enorme per

far girare tutto molto bene.

Ci sono però un paio di piccoli problemi dal punto di vista del design, però. I nemici, per cominciare, sembrano quasi tutti essere “spugne da danno” ovvero coriacei e richiedono diversi colpi per essere uccisi. Questo subito dal primo mondo di gioco e in alcuni punti può essere scoraggiante e ostico. È facile morire o essere colpiti ripetutamente senza uccidere il mostro che ci troviamo di fronte. Certo, sono presenti i continue infiniti ma diventa frustrante. Avrei preferito vedere un sistema diverso di incremento del potere delle armi e meno possibilità di continuare. Mi avrebbe dato l'apparenza di un gioco più “completo”. Ma non voglio essere duro con Silver Valley, perché è ovviamente un progetto nato per passione dal suo creatore ed è anche uno sforzo ammirevole.





GIUDIZIO FINALE

» Giocabilità 90%

Il sistema di controllo è completo e ci permette di fare tutto ciò che vogliamo. Ottima la caratterizzazione di alcuni livelli e la varietà di gioco.

» Longevità 70%

Poteva essere fatto meglio sotto il punto di vista della difficoltà e sicuramente manca di una "struttura" più solida per le armi e i colpi da infliggere ai nemici.



Se volete è disponibile la versione free da scaricare dal sito che vi ho linkato in alto nella descrizione del gioco, ma da pochi giorni è disponibile la versione fisica su cartuccia che presenta un artwork davvero ben fatto ed un ottimo manuale multilingua. Se avete la console fisica l'acquisto di questa versione vale la pena.

Concludendo Silver Valley è un prodotto tecnico impressionante sulla piccola console a 8 bit con qualche problema di design e una difficoltà sbilanciata.

di **Carlo N. Del Mar Pirazzini**



NEW GAME

KNIGHTMARE

Anno: 2022

Editore/Sviluppatore: Konami
MSX e HOFFMAN AMIGA

Genere: Shoot em up

Piattaforma: Amiga

Sito web: <https://hoffman.home.blog/>

Konami è nota per produrre giochi di grande qualità. Non ci sono dubbi su questo. La maggior parte di noi videogiocatori di vecchia data ha giocato ai titoli di questa grande azienda con grandissimo divertimento. Gradius, TwinBee, Silent Hill, Metal Gear Solid e Castelvania sono alcuni titoli che non hanno bisogno di presentazioni.

La ditta giapponese è in circolazione da molto tempo nel settore. Si erano già creati un nome all'inizio degli anni 80 con titoli come Frogger o Scramble nel settore arcade.

Nel 1986 pubblicarono un titolo chiamato Nightmare per la piattaforma MSX. Una piattaforma molto amata dalla Konami e supportata con centinaia di divertenti titoli. Quando uscì era un gioco avanzato per la sua epoca. Fu un grande successo tanto da indurre la casa di sviluppo a proseguire con un secondo titolo e un terzo.

Oggi, grazie ad Hoffman (che ha recentemente convertito il primo Metal Gear per Amiga che abbiamo recensito nel numero 31 Ita e 09 Uk) ci troviamo davanti alla versione Amiga. Il background del gioco è abbastanza semplice. Noi controlleremo un cavaliere di nome Popolon che ha la missione di salvare la Dea dell'Amore e della Bellezza, Afrodite.

La poveretta è tenuta prigioniera dal perfido Hudnos, malvagio principe delle tenebre.



Nightmare è uno sparatutto a scorrimento verso l'alto. Come in altri titoli del genere dovremo sopravvivere al continuo assalto di pericolosi nemici. Disponibili in tutte le forme possibili: pipistrelli, scheletri, slime neri, orchidee, draghi.

Per affrontarli avremo a nostra disposizione un arsenale di armi di tutto rispetto che potremo trovare lungo il gioco. Ogni arma a pro e contro, quindi è necessario possedere l'arma giusta al momento giusto. Ci sono anche potenziamenti che possiamo raccogliere durante la nostra avventura. Scudi protettivi, pozioni di invincibilità, blocca timer e tanto altro.

Al termine del livello si presenterà un malvagio boss di fine livello che potrà essere sconfitto trovando il "sistema"





giusto.

Knightmare è un gioco difficile anche in questa sua incarnazione su Amiga, come lo era su MSX. Necessita di pratica e pazienza e una buona memoria. Questa versione Amiga cosa presenta di diverso dalla versione MSX?

La grafica è stata rimasterizzata e resa più moderna e godibile da Toni Galvez. Animata con cura e molto varia.

Ci sono nuove schermate per il caricamento, è possibile opzionare il gioco a 50hz o 60hz e sono stati aggiunte alcune chicche come un nuovo bonus segreto.

Mi ha colpito molto il reparto audio. La musica e gli effetti sonori sono davvero molto belli. Un lavoro magistrale.

Il gioco si può scaricare liberamente dal sito che vi ho messo in descrizione. È in versione ADF e quindi è riproducibile su emulatore, gotek e potete caricarlo anche su dischetto e giocare su real hardware.

Funziona su tutti gli Amiga con 512kb di Ram + altri 512kb. Su A500 va che è un piacere.

È prevista anche una versione installabile nei prossimi mesi.

All'interno del file zip potrete anche trovare le artwork che ha realizzato l'autore.

Concludendo è un gioco giocabilissimo anche in questa sua versione "moderna". Lo era su MSX e rimane solido il gameplay e il divertimento anche su Amiga.

Come dicevo è difficile, ma piacevole e si lascia giocare più e più volte prima di terminarlo.

Scaricatelo e supportate l'autore, sta facendo ottimi prodotti.

di **Carlo N. Del Mar Pirazzini**

GIUDIZIO FINALE



» Giocabilità 90%

Divertente e ben strutturato. Un must su MSX. un must su Amiga

» Longevità 90%

È difficile ma vi tiene incollato al joystick come pochi.

Konami.

AMIGA



Port by Hoffman and Toni Galvez



NEW GAME

RETALIATE C.E.

Anno: 2021

Sviluppatore: Ivcabral

Musica: Richard

Editore: Ivcabral

Genere: Sparatutto

Piattaforma: Commodore 64

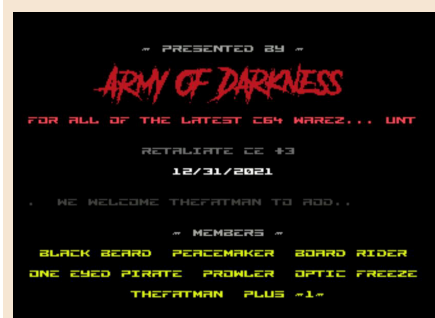
E' ormai noto che la scena videoludica per gli otto bit (e anche i sedici) è tornata a splendere come un tempo. Con cadenze sempre più ravvicinate vedono la luce nuovi giochi, conversioni e riedizioni migliorate di vecchi capolavori. In questo mi trovo a giocare e recensire RETALIATE C.E., un gioco originariamente uscito per la piattaforma Roku, uno streaming box/TV OS.

Esistevano già delle versioni per Commodore 64 di questo shoot em up fuori dalle righe: le prime versioni sono datate 2018. Nel 2019 vide la luce la versione DX, della quale troveremo un link per l'acquisto in questa edizione C.E., ovvero community edition. Originariamente uscì il 30 dicembre 2021, si segnala che il giorno seguente i mitici ragazzi della Army of Darkness rilasciano la versione craccata, dove troveremo la possibilità di rendere infinita energia, missili, e phaser.

Allora in cosa consiste questo ottimo videogame? Abbiamo tra le mani uno Shoot em Up di ottimo livello e soprattutto unico nel suo genere. A differenza dei classici stereotipi del genere, in Retaliate si inizia il gioco senza proiettili. Non nego la mia sorpresa, non avendo mai giocato alle precedenti uscite, nello schiacciare il pulsante fire e non vedere nulla uscire dall'astronave.



Il Joystick era ben collegato, tutto si muoveva ma non c'era ombra di proiettili sullo schermo. Morivo a ripetizione, fino al momento in cui l'esperienza di videogamer mi portò dapprima a pigiare tasti vari sul biscottone e, in seguito, a provare a tenere pigiato il pulsante di fuoco, quasi a cercare un'arma tipo il beam





di R-Type, ma nulla. Con stupore e meraviglia vidi che la navicella, quando tenevo il joystick in posizione indietro, venire avvolta da uno scudo che mi proteggeva da urti e spari nemici.

Contemporaneamente una linea di energia posta sotto lo schermo scendeva e lì ho capito il colpo di genio assoluto, ovvero che per andare avanti in questo gioco si deve studiare più la fase difensiva che quella offensiva. Infatti l'unico modo per sopravvivere è usare lo scudo non solo come difesa, ma anche per raccogliere i proiettili nemici, da utilizzare in seguito contro di loro. In pochi minuti il gioco prende il sopravvento, non si riesce a staccarsi da esso!

Abbiamo a disposizione una sola e misera vita ma è incredibile come, pianificando alla meglio l'utilizzo dello scudo con il posizionamento della navicella, si riesca a procedere anche quando il gioco diventa veloce. Si tende a dimenticare che abbiamo qualche colpo da sparare, ma quando lo si ricorda è perché la situazione è diventata veramente ostica. Graficamente il gioco è molto ben fatto, con un parallasse di stelle che scorrono sotto di noi, mentre

le navicelle nemiche hanno decisamente preso come esempio quelle di Galaxian e Galaga. Carina la possibilità di cambiare, nel menu principale, colore dello scudo e foggia della nostra astronave, con profili che si sbloccheranno al raggiungimento di determinati punteggi.

La musica in game ha un incidere techno ed è trascinante, mentre i sound FX ben si incastrano con essa, non dando mai l'idea di portare via una voce. In definitiva un gioco fuori dalla logica del genere che piacerà, proprio per questo, anche a chi non ama particolarmente sparare a tutto ciò che si muove sullo schermo.

di Michele Novarina

GIUDIZIO FINALE

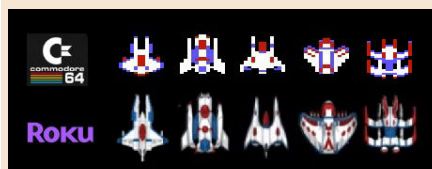


» Giocabilità 85%

Inizialmente si deve prendere confidenza con la meccanica di gioco completamente ribaltata rispetto al solito

» Longevità 95%

Una volta capita la meccanica di gioco è quasi impossibile staccarsi da esso.




NEW GAME

MEGA TURRICAN DIRECTOR'S CUT

Anno: 1994-2021

Editore: Data East/Strictly Limited

Genere: Platform

Piattaforma: Sega Megadrive

Sito: [https://](https://store.strictlylimitedgames.com/products/mega-turrican-collection-mega-drive-preorder)
[store.strictlylimitedgames.com/
products/mega-turrican-
collection-mega-drive-preorder](https://store.strictlylimitedgames.com/products/mega-turrican-collection-mega-drive-preorder)

E' da pochissimo tempo uscita la versione Director's Cut di questo classico su Sega Megadrive.

Cosa contiene? La versione originale del gioco, la versione score Attack (dove si dovrà combattere con il tempo e realizzare il miglior punteggio possibile) e la versione Director's Cut che implementa alcune migliorie grafiche. Tutto questo per 50 euro, centesimo più... centesimo meno.

Quando ero piccolo mi piaceva qualsiasi cosa fosse marchiata Turrican, passavo ore davanti all'Amiga perso nei suoi livelli e nella sua grafica colorata. Anche la versione per il C64 era davvero spettacolare.

A differenza dei giochi odierni in cui ciò che vedi è ciò che ottieni, un tempo, che si trattasse di 8 o 16 bit, avevi bisogno di aprire la tua immaginazione e fare parte del meraviglioso universo sul quale si svolgevano le imprese dell'eroe di Turrican era incredibile.



Mi immergevo davvero.

Il gioco è sostanzialmente un run & gun abbastanza lineare, ma c'era qualcosa nel franchise di Turrican da catturare la mia immaginazione al punto che mi trovai a realizzare alcune storie per una serie di giochi di ruolo moderni (GURPS, chi lo conosce?).





Mega Turrican arrivò in quel periodo dove l'Amiga stava lasciando il passo e il Megadrive era nella difficile era del fallimento 32X/lancio del Saturn, però appena arrivò lo presi. Cavolo! Turrican e il Sega Megadrive, chissà cosa salta fuori!!

È difficile esprimere a parole la mia prima impressione. Non ero affatto soddisfatto. La magia era sparita. C'erano ancora i vecchi potenziamenti e layout di gioco fantasiosi, la grafica colorata anche se ripetitiva ma mancava qualcosa... Mancava quella magia dei giochi precedenti sulle altre piattaforme. Ora la chiamerei in modo diverso e direi proprio che mancasse un gameplay accettabile. Era stato fatto un compitino per fan della serie ma senza inserire quel tocco in più da renderlo un capolavoro degno del suo nome. Intanto era terrificante la difficoltà si saltò su alcune piattaforme, chiunque abbia programmato Mega Turrican ha deliberatamente deciso di rendere difficile questa cosa. In moltissime occasioni devi calcolare al millimetro pixel per pixel... pena la morte!

Non è estranea questa cosa nei platform, ma in questa versione del gioco è sbilanciata e contrasta con il gameplay

perfetto dei capitoli su Amiga o C64. Certo ci sono una marea di potenziamenti, di armi e tanti livelli ma l'azione rimane ripetitiva e il gioco sembra non decollare mai.

Scordatevi la musica di Chris Huelsbeck, qui c'è una colonna sonora che difficilmente vorrete ricordare.

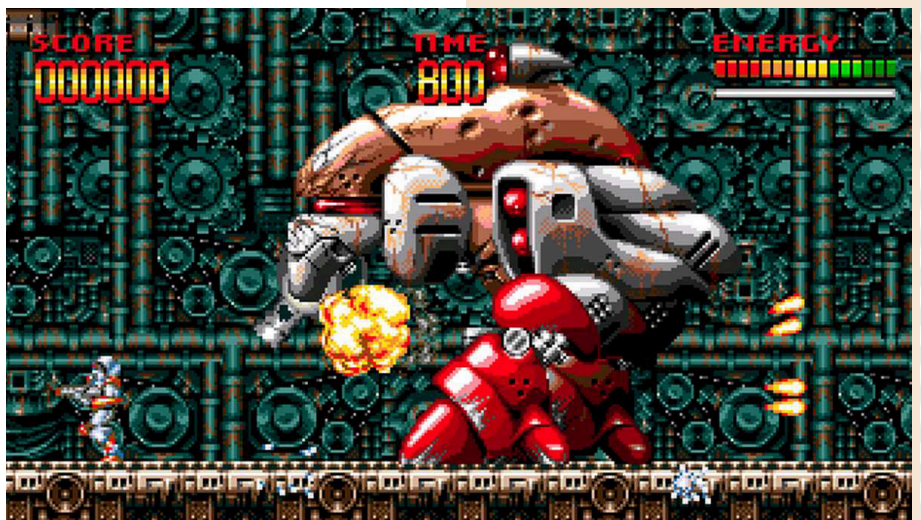
Peccato davvero. Poteva essere meravigliosamente un nuovo capolavoro. È un buon platform ma per me non è Turrican.

Questa nuova edizione aggiunge due sistemi di gioco nuovi.

La versione Director's Cut aggiunge qualcosa ma molto poco.

È mancato impegno.

di Giampaolo Moraschi



GIUDIZIO FINALE

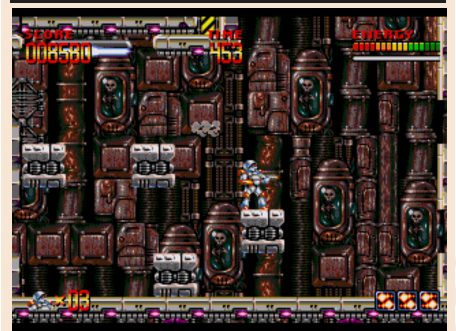


» Giocabilità 80%

Si manovra bene fino a che non si salta. In alcuni punti vi farà perdere la pazienza.

» Longevità 70%

Diverse aree molto grosse da esplorare con tre livelli di difficoltà, ma abbiamo punti dove è ultra semplice e punti dove la difficoltà cresce in maniera esagerata. Non c'è una vera curva bilanciata.




NEW GAME

EYRA, THE CROW MAIDEN

Anno: 2022

Editore: Second Dimension

Genere: Platform

Piattaforma: Sega Megadrive, Nes e Super Nintendo

Prova su: Sega Megadrive

Sito: <https://second-dimension.com/store/eyra-the-crow-maiden-sega>

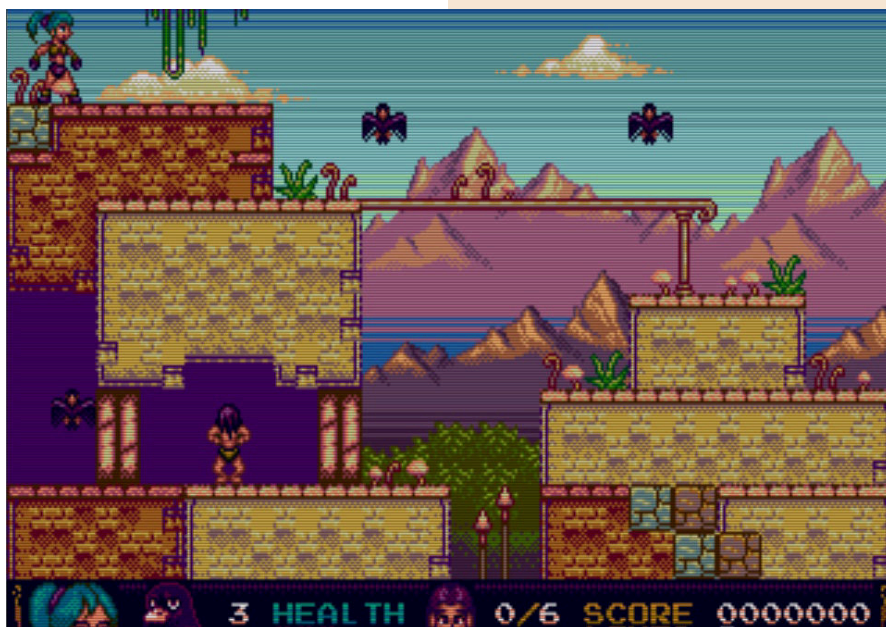
Eyra è una giovane sacerdotessa di una tribù barbara. I suoi amici e familiari sono stati rapiti dal Predone Infernale, un malvagio stregone guerriero di una terra lontana. La giovane amazzone si lancerà alla ricerca dei sopravvissuti contro orde terribili in compagnia della sua spada e del suo coraggioso corvo ammaestrato.

Si tratta di un platform game classico dove bisogna saltare, evitare trappole ed eliminare i mostri presenti, in più abbiamo il compito di liberare i prigionieri presenti nei livelli e raccogliere alcuni oggetti (rappresentati da corvi) che ci potenzieranno l'arma principale, doneranno tesori o restoreranno la nostra salute.

La Second Dimension sta lavorando davvero bene negli ultimi tempi. Non semplici homebrew senza capo né coda, ma giochi dotati di una buona trama e un ottimo comparto tecnico. Spesso accompagnati da una confezione ben fatta e da manuali realizzati con cura.

Anche questo Eyra fa parte di questo tipo di prodotti.

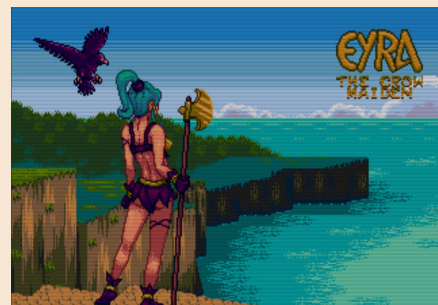
E' graficamente gradevole e ben realizzato, mi ricorda molto le produzioni Sega Megadrive europee e sono certo che un prodotto del

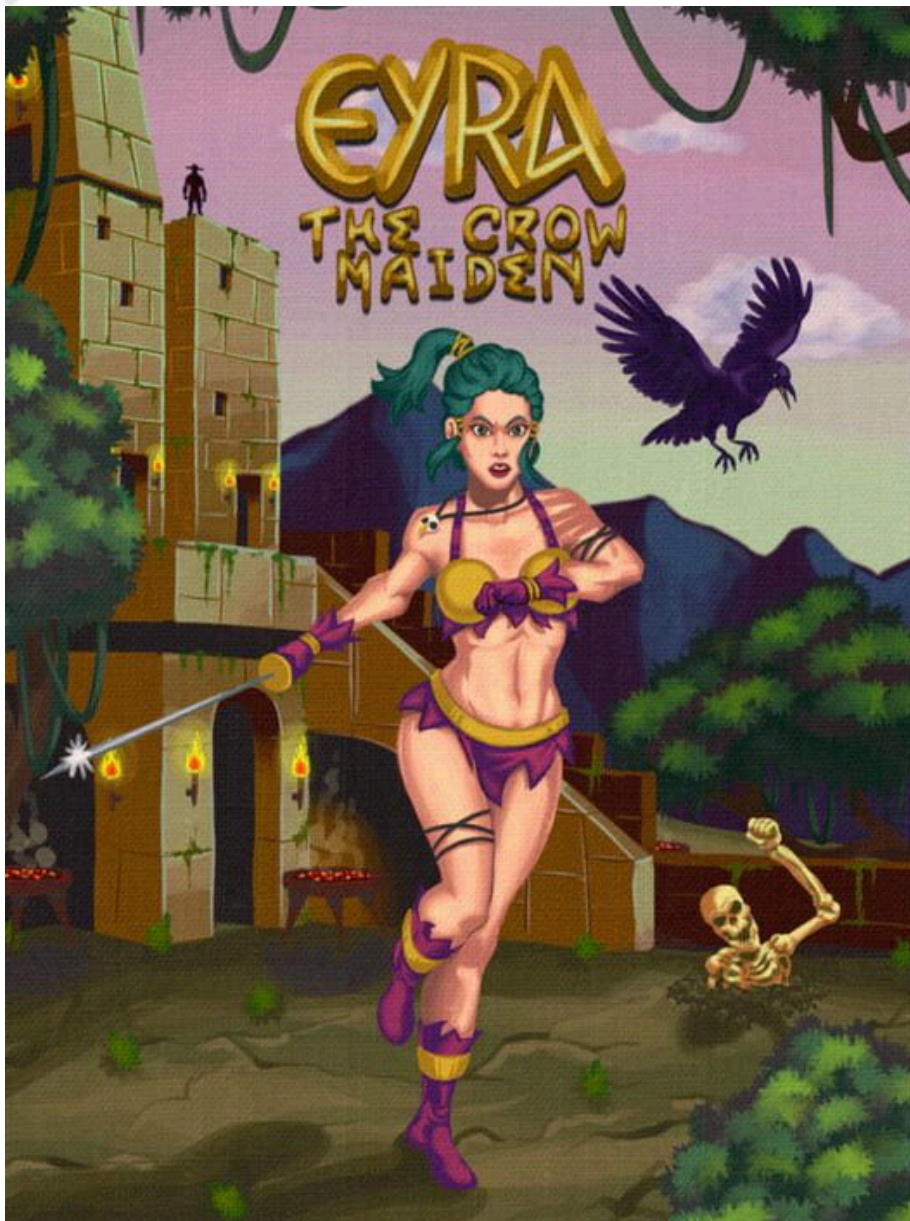


genere girerebbe degnamente anche su Amiga.

La colonna sonora è ripetitiva ma non fastidiosa.

Però abbiamo un problema...





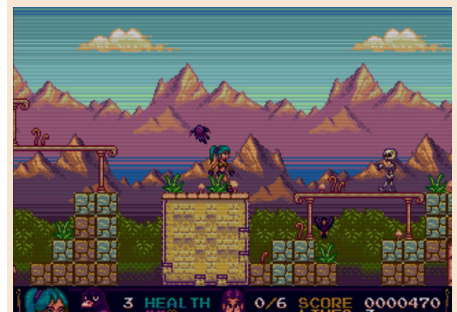
GIUDIZIO FINALE

» Giocabilità 70%

Il classico platform. Un tasto salta, un tasto colpisce, un tasto lancia il corvo. Nulla di più.

» Longevità 70%

Alla lunga può sembrare ripetitivo.



E' un gioco molto piatto. Uccidi i mostri, salta gli ostacoli, recupera i prigionieri.

Disponibile in download digitale e a breve in versione fisica.

Non che sia un male ma alla lunga diventa ripetitivo e può risultare anche noioso.

Io vi consiglio di testare il demo presente nel sito che vi ho messo nei dettagli. Se vi piace potete decidere di acquistare la Rom (come ho fatto io) o di spingervi a pre ordinare la versione fisica che sembra davvero ben fatta. Ad ogni modo è un prodotto più che sufficiente.

di **Roberto Del Mar Pirazzini**





NEW GAME

KNIGHT GUY IN LOW RES WORLD - CASTLE DAYS

Anno: 2021

Sviluppatore: Vladimir Zuniga

Genere: Platform

Piattaforma: Atari 7800

Sito web: [https://](https://vhzcgames.com/knightguycd7800.php)

[vhzcgames.com/](https://vhzcgames.com/knightguycd7800.php)

[knightguycd7800.php](https://vhzcgames.com/knightguycd7800.php)

Storia.... lineare. un drago malvagio ha rapito il nostro animale domestico. Gameplay... classico e divertente con 100 livelli da affrontare pieni di trappole e nemici.

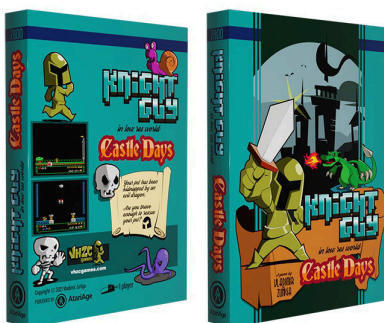
Una bella avventura per la console a 8 bit di Atari. Una console spesso passata inosservata ma che si trovò a combattere la battaglia con due colossi della sua epoca, il NES e Master System.

Come dicevamo, lo scopo del gioco è semplice. Il nostro cavaliere deve superare 100 livelli evitando tutti i pericoli presenti, recuperando armi e pozioni curative e affrontando i boss di fine livello presenti.

È da poco uscita anche la versione fisica su cartuccia, ma volendo sul sito di Atari Age è possibile scaricare la rom.

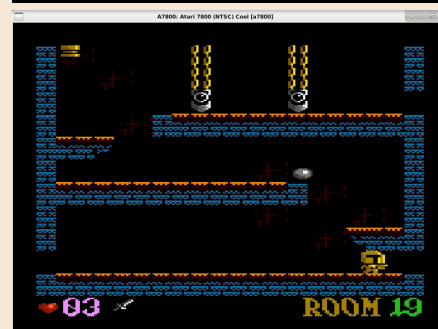
Opinione a caldo. Gioco graficamente semplice ma curato, sonoro fatto solo di effetti e una robusta giocabilità.

Consigliato agli amanti del genere platform/rompicapo. Non sarà una pietra miliare ma è molto divertente.



Lo potete acquistare in versione fisica da qui: https://atariage.com/store/index.php?l=product_detail&p=1262

di Carlo Nithaiah Del Mar Pirazzini



GIUDIZIO FINALE



» Giocabilità 80%

Semplice e molto accattivante nel design del livello.

» Longevità 80%

Non è facile e, in alcuni punti fa davvero arrabbiare, ma vi ritroverete a rigiocarlo molte e molte volte.





NEW GAME

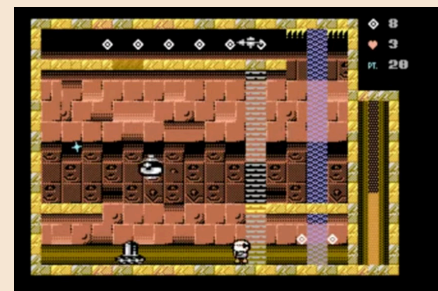
ROBOT JET ACTION

Anno: 2022

Sviluppatore: Tomasz Mielnik e Kamil Wolnikowski

Genere: Platform

Piattaforma: Commodore 64

Sito Web: <https://carrion64.itch.io/robot-jet-action>

Robot Jet Action è un classico gioco di piattaforme a schermo fisso per i nostri amati Commodore 64/128. E' stato realizzato da due super appassionati polacchi che gestiscono il portale C64Portal.pl (che vi consiglio di visitare).

Nel nostro nuovo titolo il giocatore controllerà un piccolo robot con il joystick. Cliccando sul pulsante di fuoco potrà attivare il jetpack che servirà per raggiungere tutti gli oggetti da raccogliere presenti sullo schermo. Il nostro robottino è il custode dei trofei raccolti da milioni di retrogamers nel mondo dei videogiochi del passato, ma dei malvagi alieni/robot/cattivi li stanno portando via.

Il gioco presenta una bella grafica in HI RES in modalità ECM, belle

animazioni e una colonna sonora davvero ben fatta.

Sono presenti 35 livelli tutti davvero impegnativi. Non solo dovremo evitare i nostri nemici ma anche tutti gli ostacoli presenti nel livello e dobbiamo farlo il più velocemente possibile.

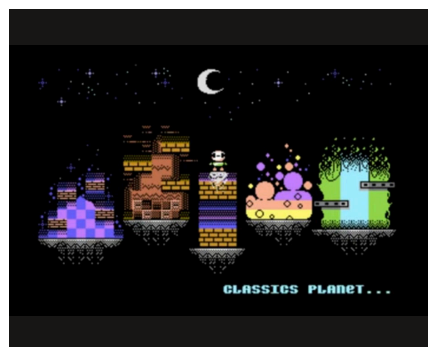
Questo tipo di titoli mi stimola moltissimo e mi portano, anche dopo essere morto più e più volte, a rigiocare il titolo.

Un bel mix tra BombJack e Manic Miner che vale davvero la pena di essere giocato.

E' gratis!!!

Supportate gli sviluppatori per i nostri retro-sistemi.

di Carlo Nithaiah Del Mar Pirazzini



GIUDIZIO FINALE

» Giocabilità 85%

Immediato e senza troppo da pensare su come muoversi. Il robot si guida perfettamente.

» Longevità 85%

Ben 35 livelli e una buona dose di difficoltà che vi terranno incollati al joystick.





SHONEN JUMP'S ONE PIECE

Anno: 2004

Editore: Bandai/Dimps
Corporation

Genere: Platform/Beat em up

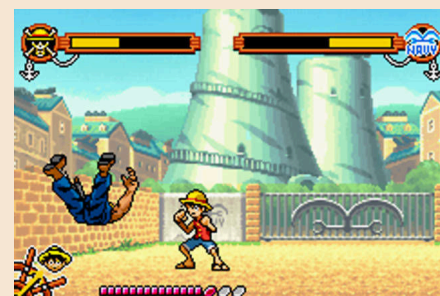
Piattaforma: Game Boy Advance

Finalmente mio! Il platform con elementi picchiaduro che più ambivo su Game Boy Advance.



senza mai infastidire. Reparto audio visivo al top.

È un gioco molto semplice e piacevole da giocare che propone una sfida di medio livello.



Ho sempre amato One Piece e i disegni di Eiichiro Oda! E questo anime per quanto sia davvero lungo non annoia mai.

Quanti pomeriggi passati al rientro dalla scuola per vedere la puntata sulla tv locale. Sempre un'emozione unica.

Passando al gioco qui guideremo Monkey D. Luffy (in Italia: Rubber) alle prese con 12 boss da affrontare attraversando 12 livelli belli pieni di nemici e pericoli vari.

Grafica coloratissima e ben animata, sonoro che accompagna l'avventura



Vi terrà impegnati per parecchio.

Non perdetelo di vista e provatelo; in real hardware o emulazione.

di **Barbara "Morgana" Murgida**



GIUDIZIO FINALE

» **Giocabilità 80%**
Si lascia giocare che è un piacere.

» **Longevità 80%**
Non troppo complicato né troppo semplice. Un buon compromesso.





GREMLINS 2 THE NEW BATCH

Anno: 1991

Editore: Sunsoft

Genere: Platform

Piattaforma: Game Boy

Nel 1984 in America esce nei cinema il primo Gremlins, film cult prodotto da Steven Spielberg e diretto da Joe Dante.

Un capolavoro del genere.

Due anni dopo, esce il seguito.

Un flop clamoroso.

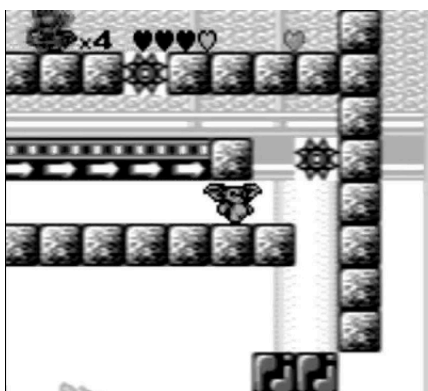
Appassionata di film anni 80/90 fin da quando ero una giovane ragazza di 16 anni, questo titolo per Game Boy non poteva mancare alla mia collezione.



Detto tutto ciò, si tratta di un platform ben curato graficamente dove il piccolo mogwai dovrà farsi strada attraverso i livelli dell'edificio dove è prigioniero, per farlo sarà dotato di armi più o meno strampalate come un super pomodoro o un arco spara fiamme.

L'obiettivo è raggiungere il centro di controllo Gremlin e spazzare via tutti i cattivi fratellastri all'interno.

Il gioco mi è piaciuto e trovo carinissimo Gizmo, mi ricorda molto i "Furby" quei buffi giocattoli di peluche

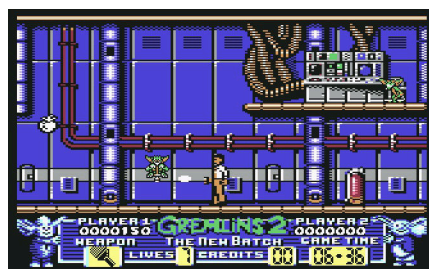


che si vendevano in Italia negli anni 90.

A proposito a voi sono piaciuti i film?

di **Barbara "Morgana" Murgida**

Altre versioni



Del gioco esistono diverse versioni su licenza per numerose piattaforme. Le versioni per Amiga, Atari St, C64, MSX, Amstrad e ZX spectrum sono sostanzialmente platform a scorrimento laterale non tutte riuscite benissimo con protagonista Billy alla ricerca di Gizmo, mentre la versione Sunsoft realizzata per Nintendo NES è un action con visuale dall'alto graficamente curato e giocabile.

di **Carlo N. Del Mar Pirazzini**



GIUDIZIO FINALE

» Giocabilità 80%

Un gioco divertente e semplice da giocare.

» Longevità 80%

Alcuni punti sono difficili ma non frustranti. Un bel titolo da passeggio su Game Boy.





CRISIS FORCE

Anno: 1991

Editore: Konami

Genere: Shoot em up

Piattaforma: Famicom/
Nintendo NES

Nel 1991 era chiaro che i giorni della gloriosa piattaforma 8 Bit Nintendo erano contati.

Nonostante la qualità dei giochi delle rivali Pc Engine e Megadrive e l'arrivo del nuovo Super Nintendo, alcuni sviluppatori non sembravano così convinti di arrendersi all'età anagrafica della piccola console.

Konami, Capcom e Enix realizzavano ancora grandissimi titoli per il Nes.

Crisis Force ne è un esempio. Uno dei più grandi shoot em up a scorrimento verticale mai rilasciato su questa console.

La storia è tipica: 199X, Tokio invasa da una forza aliena robotica. Un ragazzo e la sua ragazza decollano con i loro aerei da combattimento Falcon per fermare la minaccia.

Il gioco ha potenziamenti classici. Icona dell'accelerazione, due tipi di potenziamenti per il laser beam e una icona blu che, raccolta in quantità, rende la nave invincibile per un breve periodo.

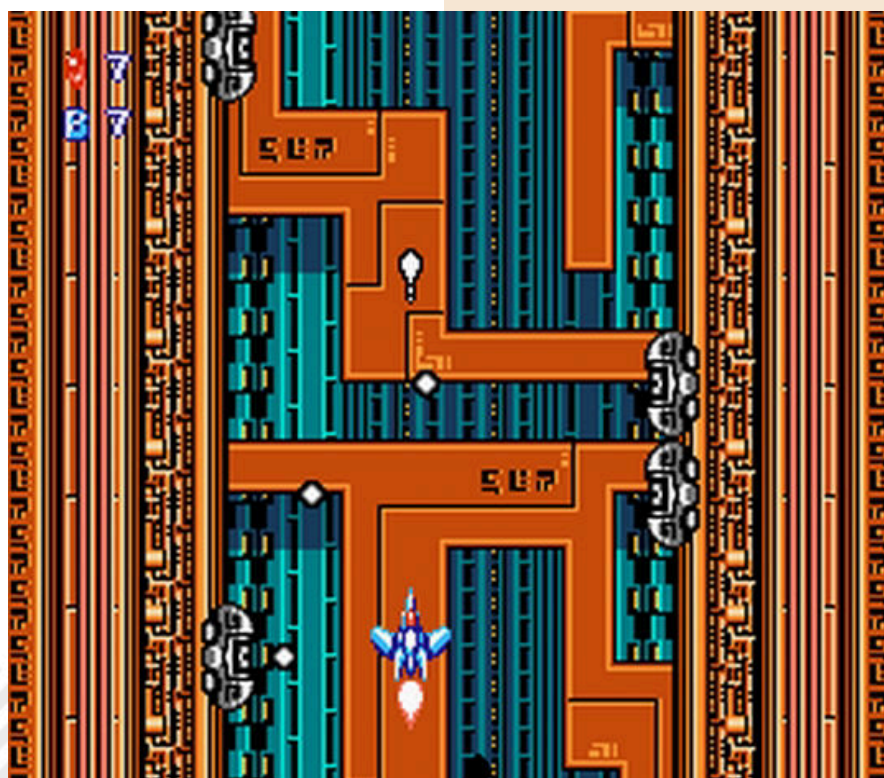
Se si sta giocando in doppio, prendendo questa icona blu è possibile combinarsi con il proprio compagni di gioco e rendere invincibile la coppia.

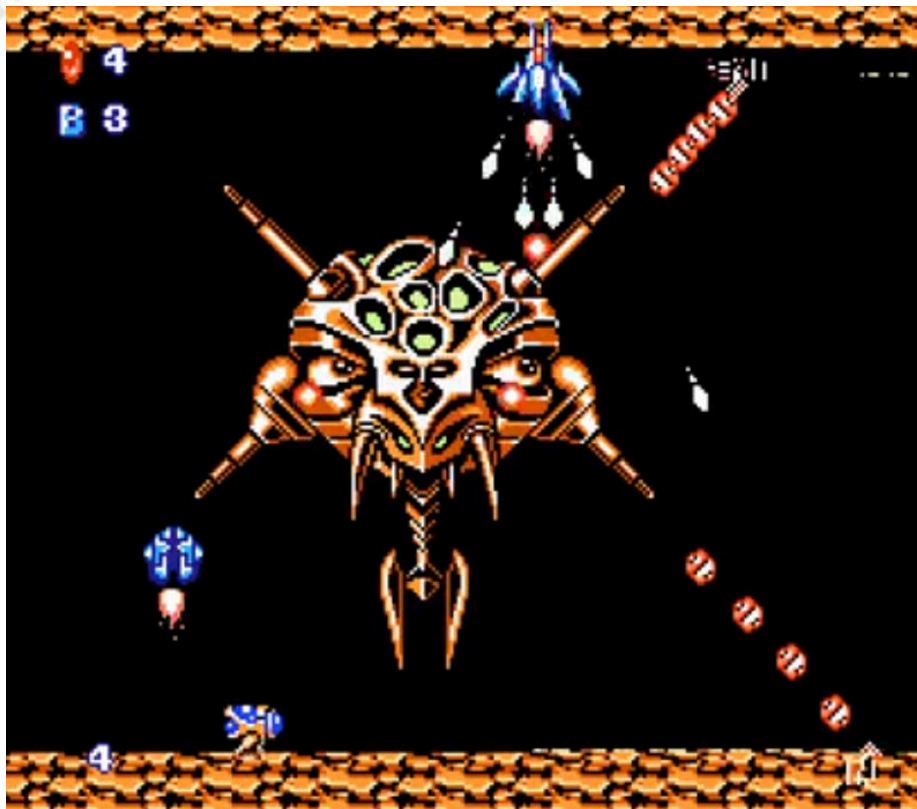
La nostra nave può anche essere trasformata durante l'azione di gioco. Premendo il tasto A si passerà dalla versione standard alla forma con un

cannone sulla parte di coda. Premendo ancora il tasto A si trasformerà in una nave con quattro torrette di combattimento ai lati.

La grafica di questo gioco è FENOMENALE! Crisis Force è stato spesso chiamato dalle riviste dell'epoca "l'AXELAY del FAMICOM", e giustamente aggiungo io.

Konami ha realizzato un livello di dettaglio e tecnica degno dei 16 bit, in termini di sfondi a scorrimento multiplo e sprite di grandi dimensioni. I chip personalizzati presenti all'interno della cartuccia hanno permesso di realizzare questa impresa. La prima volta che ho visto in azione il gioco sono rimasto scioccato. Non avevo mai visto nulla di simile sul NES/





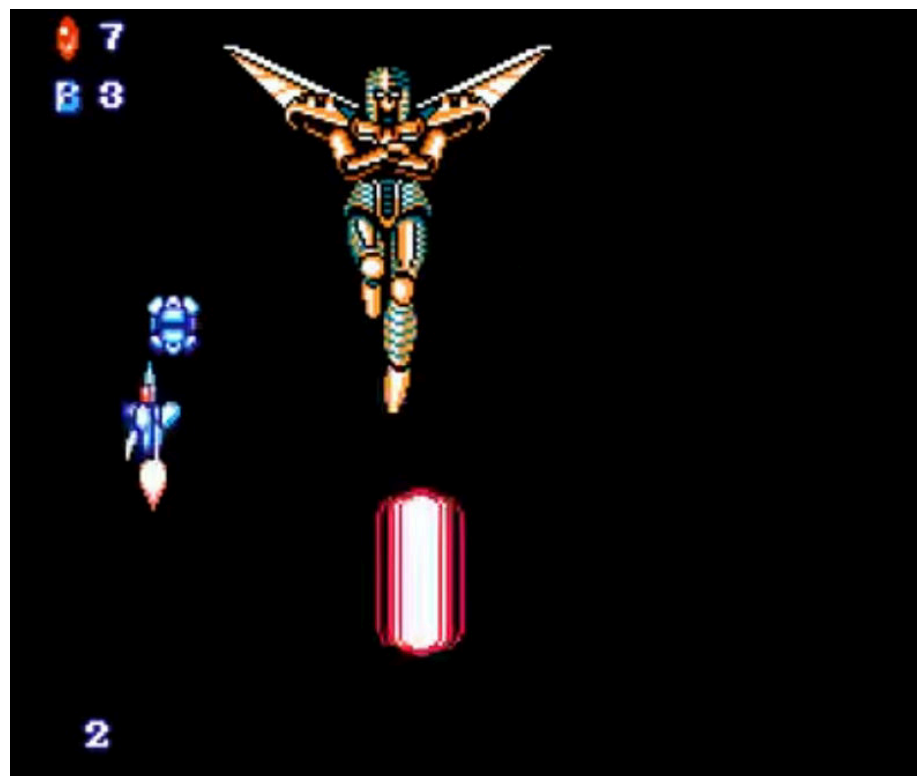
Famicom. Stesso discorso vale per il reparto audio che, in perfetto stile Konami è impeccabile e coinvolgente.

È un gioco lungo e con una curva di difficoltà ben bilanciata. È certo che bisogna cercare di avanzare nei livelli con i potenziamenti giusti, ma non è un'impresa titanica.

Crisis Force è uno sparatutto eccezionale, classificato tra i migliori giochi NES del genere. Passò inosservato sul mercato europeo ma ebbe un successo incredibile in Giappone. Curiosamente non toccò mai il suolo americano.

Provatelo!

di Roberto Del Mar Pirazzini



GIUDIZIO FINALE

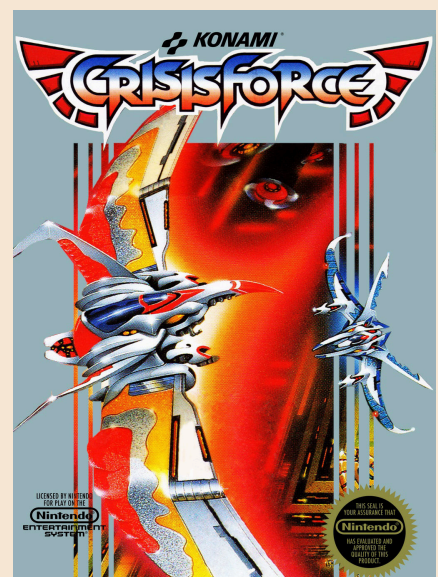


» Giocabilità 95%

È un titolo Konami. Semplice nel gameplay ma divertente. La possibilità di trasformare le navi durante il gioco lo rende accattivante.

» Longevità 90%

Sette livelli di gioco sono una bella sfida. Il livello di difficoltà è alto ma gradevole.





CLASSIC KONG

Anno: 2012

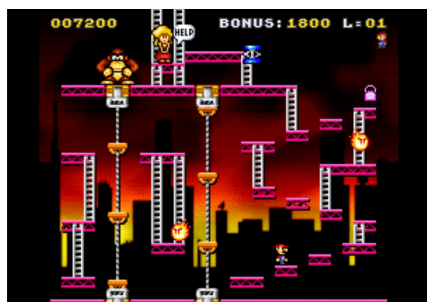
Sviluppatore: Bubble zap

Genere: Platform

Piattaforma: Super Nintendo

È un venerdì sera piovoso questo e sto giocando ossessivamente a Classic Kong che Nith mi ha passato per fare una prova.

Io l'era d'oro di questi giochi non la conosco (sono molto giovane), ma grazie alla passione di mio padre e all'amicizia di "zio" Nith mi ci sono persa. Letteralmente innamorata. Dicevo sto giocando ossessivamente a questo gioco di Bubble Zap per il Super Nintendo perché sono testarda e adoro questo titolo.



È passato molto tempo dall'ultima volta che ho preso il comando del piccolo idraulico italo-americano ed è la prima volta che mi sono cimentata con la sua più ancestrale incarnazione. Un brutto scimmione sporcaccione che ruba ragazze e le porta sulla parte più alta del livello. Povero Mario! È sempre stato sfortunato in amore.

Il sistema di gioco è quello che conoscete tutti (io in realtà da poco, ma mi sono documentata) ed è brillante, semplice e punitivo se non si fa attenzione.

Questa versione Super Nintendo è davvero gradevole. Una grafica brillante e simpatica. Che utilizza molto bene i colori della console e che sembra uscita direttamente dagli

anni 90, periodo d'oro della macchina a 16 bit.

Lo slogan "Quanto in alto puoi arrivare?" che troviamo tra un livello e l'altro fa capire quanto sia ipnotico e giocabile questo titolo. Vogliamo sempre arrivare a vedere di più e affrontare gli stessi livelli con il tasso di difficoltà che cresce in modo vertiginoso.

Sono rimasta incollata per ore e avrò imprecato tutte le lingue che conosco ogni volta che il povero Mario ci lasciava le penne. Ma sono ancora qui. Un gioco di qualità, non sembra un Homebrew di ultima generazione ma un titolo da mettere in bacheca con gli altri gioiellini per questa console. Io ho provato la Rom sul mio Super Nintendo Mini, ma Nith mi dice che gira senza problemi anche su Everdrive e che sul web si trovano anche cartucce flashate con tanto di confezione e manuale di gioco.

Buona ricerca.

di Ingrid Poggiali



GIUDIZIO FINALE



» Giocabilità 90%

Devi essere preciso a saltare, colpire e prendere il tempo giusto... ma i comandi sono perfetti!

» Longevità 95%

Il voto che ho dato parla da sé.





PROHIBITION

Anno: 1987

Sviluppatore: Infogrames

Genere: Shooter

Piattaforma: Commodore 64

Questo gioco mi ha riportato dei vaghi ricordi e di preciso di un particolare sparatutto in prima persona con mirino, sottovalutato e mai convertito per nessuna macchina casalinga: sto parlando di Empire City o più comunemente Street Fight (da non confondersi con il leggendario picchiaduro), ambientato nell'America dei gangster anni 20. Però qualcosina, il Commodore 64 si è portato a casa, anche se non ai livelli di quest'ultimo, ossia il suo clone Prohibition firmato Infogrames.

Come dice il nome, l'ambientazione è tra i quartieri malfamati americani e precisamente durante gli anni del proibizionismo. Scopo del gioco è uccidere ogni gangster che incrocia il nostro cammino o meglio, sguardo, visto che useremo il mirino di una pistola. Ad intervalli regolari comparirà uno dei gangster in ogni angolo, sui tetti, alle finestre e persino in un tombino! Tutto ciò che dovremo fare è sparargli e fino qui, potrebbe sembrare un gioco (in tutti i sensi) da ragazzi!

La schermata non è fissa come nei vecchi cabinati arcade con pistola a infrarossi, bensì a scorrimento, e la direzione da prendere è dettata dalle frecce che compaiono in basso nello schermo; noi avremo solo cinque secondi per trovare il malcapitato e riempirolo di piombo. Fortunatamente i cinque secondi possono essere rinnovati premendo la barra spaziatrice che ci farà buttare a terra per evitare la prima raffica, ma ciò ci costerà energia, raffigurata da dei proiettili. Il sonoro del gioco ricorda molto quello dei film di gangster degli anni 20, con una musica che probabilmente all'epoca andava molto, come i tormentoni estivi dei giorni nostri. E della giocabilità che dire? Muovere

il mirino di una pistola non è il massimo con il joystick o con i tasti, anche se la soddisfazione di aver fatto fuori il nemico ad un secondo dallo scadere del tempo, non è poca.

Come dicevo, il gioco è a scorrimento e si deve perlustrare un intero quartiere con nemici via via sempre più protetti: dai più impensabili ripari fino agli ostaggi! Come in Empire City, c'è un bandito che più avanti si fa scudo con una bella donna; questa sicuramente è la scena più mi è rimasta impressa del coin op originale.

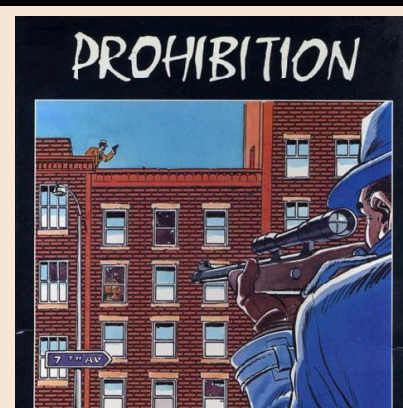
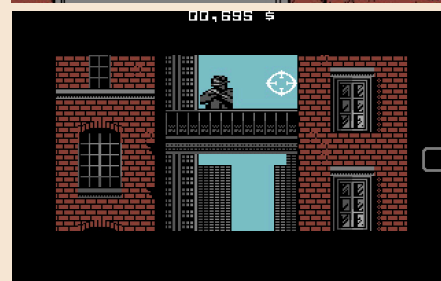
È uno di quei giochi che, anche una volta arrivati fino in fondo, una partitella ogni tanto è un toccasana, soprattutto dopo una giornata lavorativa stressante, per sfogarsi. Proprio oggi mentre lo provavo, temevo potesse essere ripetitivo e invece ho notato delle variazioni di nemici e ambientazioni andando avanti.

Il caricamento su cassetta non è neanche lungo, giusto il tempo per un caffè o un tè o magari un bel bicchierino anche se non sarebbe molto indicato dato il titolo.

Ultima cosa ma non per importanza: Infogrames è un marchio che ha sempre prodotto buoni titoli, anche se non molti hanno avuto i riflettori puntati e, come ogni gioco che ha ricevuto poche attenzioni, Prohibition si merita una seconda possibilità.

Noi cerchiamo sempre il lato positivo, perché anche nel buio più fitto uno spiraglio di luce c'è sempre! Dopo quasi due anni di pandemia e una lenta seppur significativa ripresa, auguro buona guarigione a chi è in quarantena e di tornare a vivere normalmente.

di **Daniele Brahimi**



GIUDIZIO FINALE



» Giocabilità 70%

Uccidere gangster ad un secondo dalla fine è soddisfacente!

» Longevità 60%

Non lungo, ma una partitina ogni tanto farà bene.





La diffusione dei social network ha senza dubbio riaperto tante passioni della nostra infanzia e tra queste c'è anche quella per i retrogame.

Infatti il web è pieno di pagine, gruppi e forum dedicati ai capolavori delle sale giochi e delle varie console casalinghe.

Ma mentre la maggior parte di queste realtà si limitano a far riemergere ricordi e aneddoti di queste gemme del passato o al massimo a spiegare come poterle rigiocare in single player, pian piano si è fatta sempre più strada Retro Multiplayer, una community che invece ha come scopo quello di poter rigiocare quei titoli in modalità versus o cooperativa, proprio come quando andavamo nelle sale giochi o a casa di amici.

Retro Multiplayer nasce circa quattro anni fa, quando un piccolo gruppo di appassionati di videogame crea un server discord e comincia con tenacia a sperimentare come poter rigiocare i titoli arcade in multiplayer online. Prova dopo prova, successo dopo successo, quel piccolo gruppo si è cominciato ad allargare, trovando tanti altri utenti che sono rimasti affascinati e coinvolti dal progetto.

Oggi il server discord conta più di cinquecento iscritti ed è stato affiancato da un gruppo e pagina Facebook, un profilo Instagram e Twitter e un canale Youtube e Twitch sui quali sono stati caricati più di cento video tra tutorial e serate evento.

Infatti gli utenti di Retro Multiplayer, oltre a ritrovarsi la sera per rigiocare insieme i retrogame, dal 2020 hanno iniziato ad organizzare eventi a tema trasmessi puntualmente sui vari canali social.

Tra queste serate ci sono tornei, party, quiz musicali, retrointerviste alle realtà più diffuse del retrocomputing e retrogaming italiano e naturalmente partite in cooperativa.

Inoltre da pochi giorni il team di Retro Multiplayer ha rilasciato una console gratuita con i vari emulatori settati in ogni particolare, in modo da rendere l'esperienza del multiplayer online estremamente user friendly.

Si possono così giocare titoli che vanno dai classici arcade e console a 16 bit fino alla Nintendo 64, PSP, Playstation, Dreamcast e Gamecube, oltre a diverse pietre miliari per PC.

Se avete voglia di respirare di nuovo l'atmosfera delle sale giochi e dei pomeriggi a casa di amici, vi basta solo entrare nel server discord seguendo i link che trovate in questa pagina.

Incontrerete così una community di appassionati che è praticamente diventata una famiglia che si sente e si incontra quotidianamente per condividere e rivivere insieme la passione interminabile per il magico mondo del retrogaming.

di Querino Ialongo

Anno: 2018

Piattaforma: Discord

Link server: https://discord.io/retro_multiplayer

Link Social: <https://linktr.ee/retromultiplayer>



Alcune locandine di eventi organizzati dalla community





Prima della chiusura del numero, Giuseppe ci ha inviato un'altra bellissima immagine, e sarebbe stato un delitto non condividerla con voi. Artwork di **Giuseppe Mangini** ispirato al gioco **Knightmare**.



Quattro passi dopo il pasto...

C'è stato un tempo in cui le informazioni non mi bastavano mai, un tempo in cui avrei pagato ben più di quanto ho effettivamente fatto (in edicola, in libreria...) pur di avere la possibilità di guardare oltre l'orizzonte e capire cosa nascondesse. Vivevo in un piccolo paese della provincia di Verona, oggi centro nevralgico di molte attività ma, ora come allora, troppo piccolo per fare "massa critica", nel senso che tra i tutti i miei amici non ve n'era uno che avesse il mio stesso computer: io possedevo un C128, un altro lo Spectrum, un altro ancora era passato dal VIC-20 ad un IBM con processore 8088, un altro ancora stava per mettere le mani su un Amiga 500. Tutte cose bellissime ma, di fatto, non relazionabili tra loro. Di conseguenza, a dispetto di tutti i nostri tentativi, nessun progetto comune è mai sbocciato. Purtroppo, nonostante ciò che diceva Thomas Merton, ognuno di noi era, di fatto, un'isola.

Per anni, in tempi moderni, ho cercato di tornare indietro per recuperare le occasioni "perdute", complice anche il ritrovato amore di molti appassionati per il retrocomputing e l'enorme disponibilità di materiale che allora, come detto, era quasi introvabile, soprattutto in lingua italiana. Qualcuno ha detto che, se avesse avuto un computer, Einstein avrebbe forse costruito una macchina del tempo; nel mio piccolo, se avessi avuto un paio di libri in più, certamente non avrei costruito un tale aggeggio, ma un gioco ed essa ispirato forse sì. La "fame" di sapere era così forte che in tempi recenti, prima di tornare effettivamente a programmare sugli 8-bit, ho avuto una serie di sogni ricorrenti che ho successivamente documentato sul blog con il quale collaboro. Oggi quei sogni non ci sono più, perché nella realtà esiste RetroMagazine World, un posto meraviglioso dove sperimentare mille cose, non solo col mio vecchio (e tuttora funzionante) Commodore C128, ma anche con qualsiasi sistema ad 8 e/o 16 bit con il quale mi venga voglia di lavorare. Preservazione del software, divulgazione dell'hardware, creazione di una conoscenza strutturata e condivisa, stimolazione della curiosità, integrazione, cooperazione e chi più ne ha più ne metta, questo è il mondo di RetroMagazine World.

Non è però l'unico mondo possibile. Decine di altri hobbies, ugualmente stimolanti, bussano ogni giorno alla mia porta: dal modellismo allo studio delle lingue, dai fumetti (di ogni tipo), all'animazione giapponese. E così, anche per me, le cose ogni tanto vanno fuori fuoco. Qualche giorno fa, ad esempio, mi sono ritrovato a fare una domanda ai colleghi della redazione ed ho ottenuto la seguente risposta: "Guarda che ne abbiamo parlato sulla rivista poco tempo fa!". Ed allora, come i malati in "Johnny Mnemonic", sono soggetto anch'io al sovraccarico di informazioni ed al loro consumo solo in pillole? Possibile che la "fame" che avevo sia stata saziata al punto di farmi diventare pigro?

Una volta avevamo più fame e questo aguzzava l'ingegno. Ora è meglio che anch'io, metaforicamente, esca a fare quattro passi. Dopo aver ritrovato il focus, ne sono certo, anche la "fame" tornerà.

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 6 - Numero 35 - FEBBRAIO 2022

Direttore Responsabile

Francesco Fiorentini

Vice Direttore

Marco Pistorio

Coordinatore Redazione/Editing

David La Monaca

Responsabili Area Web

Flavio Soldani/Giorgio Balestrieri

