

# Jurassic News

Western Digital WD-90

PET Terapy

La nascita dell'INTEL 4004

I primi 20 anni della PlayStation

Technosphaera - Prolog - TAMC  
Retrocomputing - Mediateca

*Retrocomputer Magazine*

Anno 10 - Numero 53 - Aprile 2015

## **Jurassic News**

*Rivista aperiodica di Retrocomputer*

*Coordinatore editoriale:*

*Tullio Nicolussi [Tn]*

*Redazione:*

*redazione@jurassicnews.com*

*Hanno collaborato a questo numero:*

*Lorenzo [L2]*

*Salvatore Macomer [Sm]*

*Besdelsec [Bs]*

*Lorenzo Paolini [Lp]*

*Giovanni [jb72]*

*Antonio Tierno*

*Piero*

*Diffusione:*

*Lettura on-line sul sito o attraverso il servizio Issuu.com; il download è disponibile per gli utenti registrati.*

*Sito Web:*

**[www.jurassicnews.com](http://www.jurassicnews.com)**

*Contatti:*

*info@jurassicnews.com*

*Copyright:*

*I marchi citati sono di copyrights dei rispettivi proprietari.*

*La riproduzione con qualsiasi mezzo di illustrazioni e di articoli pubblicati sulla rivista, nonché la loro traduzione, è riservata e non può avvenire senza espressa autorizzazione.*

## **Jurassic News**

***promuove la libera circolazione delle idee***

## **Jurassic News**

*E' una fanzine dedicata al retro-computing nella più ampia accezione del termine. Gli articoli trattano in generale dell'informatica a partire dai primi anni '80 e si spingono fino ...all'altro ieri.*

*La pubblicazione ha carattere puramente amatoriale e didattico, tutte le informazioni sono tratte da materiale originale dell'epoca o raccolte su Internet.*

*Normalmente il materiale originale, anche se "giurassico" in termini informatici, non è privo di restrizioni di utilizzo, pertanto non sempre è possibile riportare per intero articoli, foto, schemi, listati, etc..., che non siano esplicitamente liberi da diritti.*

*E' possibile che parti del materiale pubblicato derivi da siti internet che non sono citati direttamente negli articoli. Questo per la difficoltà di attribuzione del materiale alla fonte originale; eventuali segnalazioni e relative notifiche sono benvenute.*

*La redazione e gli autori degli articoli non si assumono nessuna responsabilità in merito alla correttezza delle informazioni riportate o nei confronti di eventuali danni derivanti dall'applicazione di quanto appreso sulla rivista.*

*Aprile... dolce dormire.*

*Ma noi non dormiamo certo sugli allori :-)  
Ed ecco a voi un nuovo numero della rivista specializzata, e  
vorrei dire ossessionata, dalla vecchia informatica.*

*Sempre qualcuno mi chiede il perché del mio interesse per  
“...i computer vecchi che ormai non servono a niente!”,  
come dice “il saggio”.*

*In realtà la retro-informatica si occupa parzialmente  
di macchine cioè dell’hardware, il focus è sulla Cultura  
Informatica (CI) in generale, ma è un concetto difficile da  
spiegare, quasi quanto il mistero della Trinità!*

*Alla fine è vero: ci sono “le vecchie glorie”, intese sia come  
sistemi di calcolo che come programmi, ma è “il tutto” che si  
deve studiare, analizzare e infine preservare.*

*Cioè il sistema, le sue periferiche, il suo software di base, i  
suoi programmi, la sua documentazione....*

*Ascoltare che “... il personal computer è nato in Italia dalla  
Olivetti...”, frase che mi è capitato di ascoltare più di una  
volta nei programmi radio e televisivi, è (forse) vero ma  
incompleto.*

*Va bene, c’è la perottina... ma mica basta!*

*Chi fa affermazioni così semplicistiche non intende  
approfondire il contesto, oppure lo dice per sentito dire.  
Accetto l’affermazione da chi riesce a spiegarmela, magari  
non tecnicamente ma culturalmente;  
viceversa sono solo parole.*

*Ecco dove diventa importante il nostro “lavoro”: far  
percepire agli altri la vera natura dell’evoluzione culturale  
che abbiamo vissuto e che ha portato alle macchine di oggi,  
non semplicemente conservare “la scatola”, magari pure  
vuota...*

*Buona lettura.*

## L'editoriale

4

*Aprile... dolce dormire*

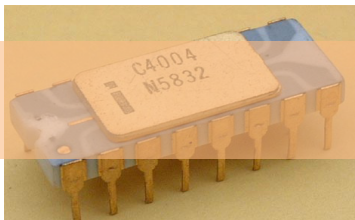
8

*Prolog (parte 6)*



*Retro linguaggi*

18



*Come eravamo*

*La nascita dell'INTEL 4004*

## TI Club

24

*Francomputer - Tutto TI-99*



26



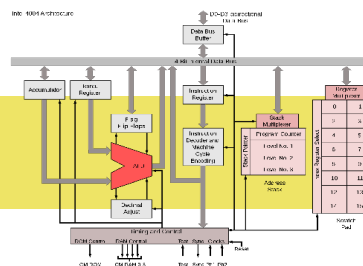
*Prova hardware*

*Western Digital - WD-90*

## Retro Code

26

*Programmare l'INTEL 4004*



52

*Come eravamo*

*I primi 20 anni della PlayStation*



## Retrocomputing

*L'oggetto del nostro scontento*



6

## Laboratorio

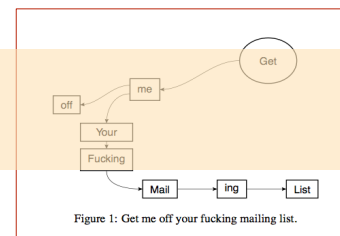


*Pet therapy*

10

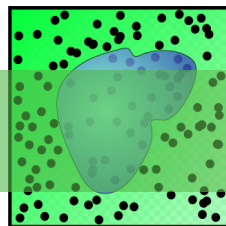
## TechoSphera

*Take me off...*



22

## TAMC



*Il metodo Montecarlo*

34

## Il racconto

*Automatik (23) - Il lato oscuro*



38

## Mediateca

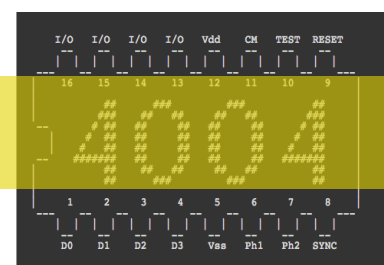


*La zampata del vecchio leone  
The rise of DDoB's*

46

## Emulazione

*Sistema che vai...*



50

## L'oggetto del nostro scontento



di Tullio Nicolussi

*Il collezionista è una persona strana: persegue ciò che magari altri per avventura fuggirebbero.*

*Taluni considerano la passione del retro computing una bizzarria. “Che senso ha” - dicono - “mettersi in casa un catorcio ormai nemmeno buono per farci il sudoku?” Quando poi gli si informa del fatto che non solo piace “quel catorcio” ma ci lavoriamo anche e amorevolmente lo accendiamo ogni tanto con trepidazione e andiamo ad ascoltare con attenzione che non ci siano rumorini strani o difetti sul video o un incerto apparire dei caratteri digitati sulla tastiera...*

*Quando poi la macchina si guasta, e inevitabilmente succede, allora sono piccoli o grandi drammi, soprattutto se succede “proprio a quella macchina lì...!”. Perché, lo sappiamo tutti, ci sono le “macchine favorite”, quelle che ognuno di noi preferisce alle altre per imprintig magari, per un qualche altro accadimento della vita o semplicemente perchè la si è tanto desiderata e non la si*

*poteva avere.*

*Il collezionista ha sempre qualcosa di mancante nella sua collezione. E’ proprio rarissimo che qualcuno possieda una serie completa e non più ampliabile di oggetti.*

*Qual è la macchina del vostro scontento? Quel sistema che vorreste avere a tutti i costi (o quasi) e per il quale vi sembra che la vostra collezione non valga quanto dovrebbe?*

*Facile sentire dell’Apple-1: bella forza! A parte che se lo possedessi l’avrei già venduto da un pezzo! Penso che non abbia senso possedere un oggetto il cui valore venale vada oltre le proprie possibilità economiche: sono solo preoccupazioni!*

*Io sono passato attraverso fasi per le quali la macchina desiderata è cambiata nel tempo. Lasciando perdere i desideri irrealizzabili per varie ragioni, come quello di avere un PDP/11, ho desiderato ad un certo punto una Sun/120. Proprio mi ero fissato e probabilmente il motivo è perché su quel sistema ci ho fatto la tesi di laurea. Forse il desiderio di*

un ritorno al passato... Peraltro forse nessun collezionista o quasi come chi si occupa di retro computer desidera un ritorno "ai bei tempi". Un altro sistema "must have" fu l'HP-85; anche su questo ci avevo lavorato, anzi posso dire di averci realizzato dei programmi di analisi di misure di laboratorio veramente cool (per l'epoca)!

Poi c'è stata la fase dello Spectrum. Devo premettere che sto parlando degli anni a cominciare dal 1990, quando di retro computing non si parlava proprio ma c'erano in giro solo "vecchi computer"...

Io non avevo avuto lo Spectrum ma è sempre stata una macchina che per fama o per altri fattori mi sarebbe piaciuta. Me lo procurai facilmente in un mercatino, così come il Commodore64, altra macchina mai avuta ma che dovevo possedere per fare alla pari con la creatura di Sinclair.

Però quella era la fase del "lo prendo per usarlo", non per aggiungere un pezzo alla collezione.

Poi il Mac "compatto" mi sembrò indispensabile e via via a catena passando per l'AS/400 e altri sistemi "impegnativi". Poi è stata la volta dell'innamoramento delle macchine dell'ingegnere Becattini: i vari General Processor ma anche le schede sperimentali progettate prima e illustrate sommariamente su CQ a beneficio dei pochissimi allora che ci capissero qualcosa.

Poi fu la volta di una macchina particolare: Black Star; un sistema CP/M costruito a Trento, cercato e ricercato con il lumicino fino ad arrivare al progettista in persona che manco a dirlo non aveva più nien-

te avendo cambiato lavoro più volte nel corso degli anni.

A volte mi sono "passati sotto il naso" sistemi che non suscitavano il mio interesse ma che poi mi sono pentito di non aver preso. Mi viene in mente l'Aquarius: ne trovai addirittura due con scatola e tutto il resto in un mercatino e ricordo che il signore che li cedeva a poche decine di migliaia di lire disse di averne comprato due uguali per le sue figlie gemelle.

Lo consideravo un giocattolo "alla Barbie" e non lo presi. Certo, potrei procurarmelo anche oggi pagandolo su eBay, ma non è questo il punto.

Poi questa smania del possesso mi si è placata come d'incanto. Ho tenuto poche macchine e non ne cerco di nuove. Anzi, alcuni sistemi li ho regalati o ceduti a poco quando mi sembrava di cogliere in chi me li chiedeva quel "sacro fuoco" che alimenta chi va cercando proprio quel sistema!

Certo se avessi un Apple-1... :-)

(\*)



# Prolog (Parte 6)

## Finale



*di Salvatore Macomer*

Il nostro viaggio nel linguaggio Prolog si conclude con questa puntata. Ovviamente il percorso non è stato esaustivo, voleva essere un approccio ad un linguaggio inusuale rispetto alle normale logiche programmatorie che si sono imposte nell'ultimo decennio.

Con questo non vogliamo dire che Prolog sia inferiore o sia abbandonato, anzi! Esistono fior di implementazioni e in alcune nicchie viene ancora usato.

Per concludere con una certa completezza la panoramica sulle caratteristiche del linguaggio ci manca un operatore molto importante, l'operatore "cut" che viene rappresentato dal simbolo del punto esclamativo "!".

Come dice il suo nome, l'operatore cut "taglia", cioè viene usato per spezzare la valutazione in sequenza dei goal nelle clausole.

Facciamo un esempio.

Abbiamo già programmato in una puntata

precedente l'operatore che cerca all'interno di una lista le occorrenze di un certo elemento.

```
member(X,[X|L]).
```

```
member(X,[Y|L]) :- member(X,L).
```

L'implementazione, la ripetiamo per riprendere il ritmo del discorso, è la seguente:

X è membro della lista data come secondo argomento se è la testata della lista (prima clausola), oppure se è membro della lista formata da ciò che rimane della lista togliendo il primo elemento.

La valutazione ad esempio di:

```
? :- member(a, [c, d, e, a, f, a, z]).
```

fornisce due risposte positive perché l'elemento "a" è contenuto due volte nella lista.



Il meccanismo di valutazione del Prolog infatti cerca di risolvere la totalità dei goal associati.

Se volessimo una sola risposta indipendentemente dal numero di occorrenze dell'elemento "a" nella lista dovremmo fermarci al primo elemento trovato.

Usando il cut questo è banale. Riscriviamo la prima clausola:

```
member(X,[X|L]) :- !.
```

Il simbolo "!" (che significa "cut", "taglia") dice al meccanismo di valutazione che se viene soddisfatta questa clausola, cioè è stato trovato l'elemento perché in testa alla lista, allora la valutazione si chiude perché "!" rende la clausola falsa.

Vediamo un'altro esempio. Supponiamo di avere un programma così definito da due clausole:

```
x :- p, !, q.  
x :- r.
```

La prima clausola procede valutando p, se è falsa fallisce e sarà considerata la seconda, cioè r.

Se invece p è vera il meccanismo procede e incontra il cut "!" a quel punto non può tornare indietro e si hanno due casi: o q è vera e allora x sarà valutato a q, oppure è falso e la clausola fallisce del tutto.

In un linguaggio procedurale sarebbe qualcosa come:

```
if p è vera ==> allora q, altrimenti r.
```

### Il mito della correttezza.

Una delle questioni più dibattute nell'ambito della teoria dell'informazione è il tentativo di definire formalmente la correttezza di un algoritmo. Un linguaggio di programmazione è un "Linguaggio" a tutti gli effetti, anche se limitato come numero di termini e come regole. Ad esempio non ci sono le forme verbali del passato e del futuro.

Nel linguaggio naturale non è difficile individuare dei paradossi, cioè frasi che non risultano né vere né false. La domanda è: "Può accadere lo stesso per un linguaggio di programmazione?".

Quando il Prolog come linguaggio si affacciò al mondo degli sviluppatori non accademici, si diffuse una falsa credenza: non è possibile scrivere programmi in Prolog che siano formalmente errati.

Il ragionamento era molto semplice: dal momento che Prolog implementa il ragionamento logico, se la logica è corretta (e lo è dai tempi dei filosofi greci), allora ogni programma sintatticamente corretto è formalmente corretto.

La realtà non è questa, ovviamente e la rottura di questo mito fece perdere parecchi punti al Prolog nella stima dei programmatori e degli stessi accademici. Anzi si è sviluppato tutto un filone di ricerca che andava scavando tutte le idiosincrasie del linguaggio!

Parleremo di Logica (con la L maiuscola perché la merita) uno di questi giorni...

(\*)

# Pet Therapy



Di Piero

## **Come ti rimetto in funzione il PET 2001.**

Il precedente proprietario non l'aveva nascosto: quel PET 2001 modello 1977 "prima serie" non era funzionante. A parte essere completo e originale, il computer si presentava davvero male, con parti interne svitate e sporche. Una macchina che per anni aveva svolto il suo lavoro e poi era finita in cantina, utilizzata saltuariamente come tana o nascondiglio dalla fauna locale. La prima operazione non evitabile è la pulizia più drastica, estesa alla piastra madre con immersione in acqua tiepida e detersivo prima d'essere accuratamente spazzolata. I terminali dei chip e i connettori appaiono sporchi e ossidati e per questo si ricorre all'applicazione dell'apposito spray a secco per la pulizia dei contatti seguita dall'azione meccanica di spazzolamento e movimento per ravvivare i punti di contatto. Anche la tastiera ha bisogno di cure. E' smontata dal

PET e tenuta a rovescio mentre una spazzola in acqua saponata tenta di liberarla fa anni di sporco evitando che coli all'interno. Infine è usato l'alcool per togliere le tracce di sporco dalla contattiera interna dei tasti.

Una dote fondamentale per chi voglia recuperare vecchi computer è la pazienza, posta a freno dell'impulso di dare subito tensione che si risolverebbe facilmente in una nuvola di fumo. Prima di tutto occorre testare l'alimentatore e, in questo caso, procedere alla rigenerazione del grosso condensatore elettrolitico che è tra le componenti principali. La rigenerazione serve a ripristinare lo strato dielettrico (isolante) che la mancanza d'utilizzo fa degradare. Il condensatore è staccato dal circuito, collegato (rispettando le polarità) in serie a una piccola lampadina e sottoposto a tensioni gradualmente crescenti fino alla sua tensione nominale. Questo in un tempo di circa 24 ore, attendendo che l'assorbimento del condensatore si riduca a livelli minimi. L'aspetto dei condensa-

tori "a goccia" al tantalio della scheda madre suggerisce di procedere alla loro sostituzione precauzionale. Se alterati questi piccoli condensatori hanno il vizio di andare in corto circuito e combinare danni.

Estratti i componenti su zoccolo dalla scheda e riposti in ordine su un supporto in polistirolo e foglio d'alluminio (precauzione antistatica che affianca quella di connettere la scocca del computer a massa e toccarla prima di maneggiare i chip) si procede a misurare le tensioni di alimentazione. Prima d'inserire la spina si controlla il fusibile, che si rivela di valore sbagliato e tale da non offrire alcuna protezione. Acceso il computer, le letture della tensione con il tester a monte e a valle dei regolatori appaiono corrette. Si rimontano quindi gli integrati accertrandosi con le dita che il riscaldamento sia nella norma.

Poiché il PET non dà segnali di vita si comincia la ricerca dei guasti dal monitor, che ha il CRT con il filamento acceso, l'alta tensione (la statica è percepibile con il dorso della mano sullo schermo) ma luminosità pari a zero. La situazione che non cambia agendo sulla regolazione della luminosità. Una verifica delle tensioni sulla scheda monitor rivela un guasto a un diodo zener che stabilizza a 33V la tensione di alimentazione dello stadio di amplificazione video. Smontata la scheda del monitor si sostituiscono anche due condensatori elettrolitici sui rami che alimentano le tensioni di griglia. La luminosità ritorna, disegnando una riga orizzontale.

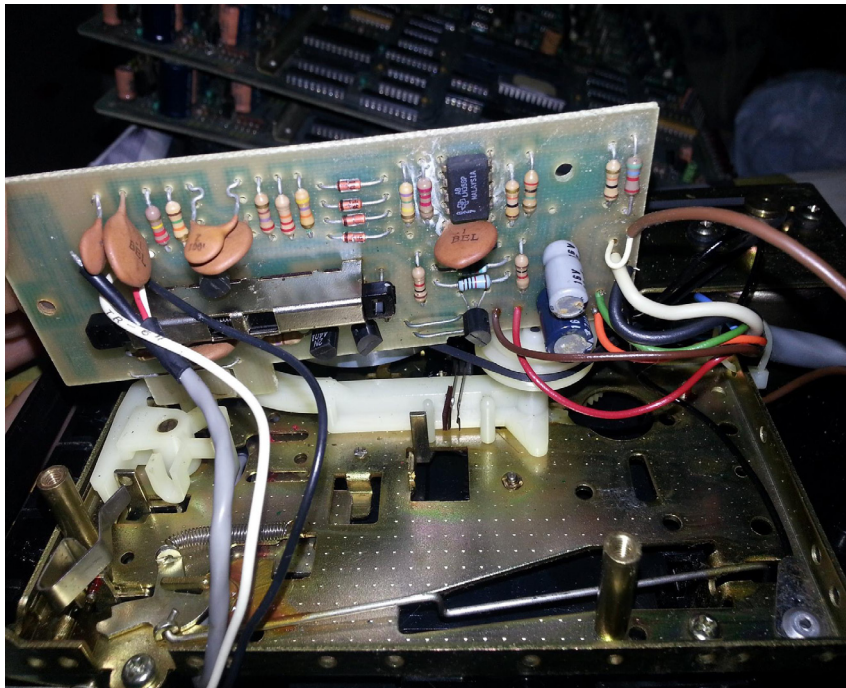
Quando un monitor CRT visualizza una

riga orizzontale significa che manca la deflessione verticale, di solito a causa del distacco dei fili che vanno alle bobine sul collo del CRT oppure di un guasto al relativo circuito. Sul vecchio PET a mancare è il segnale verticale che proviene dalla scheda madre. L'oscilloscopio ci rivela che al posto del 60Hz previsti ci sono treni di impulsi a 250KHz. Il guasto insomma non è nel monitor.

### **La riparazione del circuito video.**

Lo schema elettrico ci rivela i tortuosi percorsi con cui il PET ricava il sincronismo verticale a partire dagli 8 MHz generati dal quarzo presente sulla scheda, lo stesso da cui ricava anche il clock da 1MHz della CPU. Misurando le frequenze all'ingresso e all'uscita dei vari circuiti si scopre che due integrati 74LS177 coinvolti nella scansione della RAM video sono guasti, sono quindi dissaldati e sostituiti. Il segnale del sincronismo verticale ritorna normale. Rimettendo al loro posto la RAM video e la ROM dei caratteri ecco finalmente apparire il quadro con una schermata piena di chiocciole. Rimontando anche la CPU ecco che appare la schermata di caratteri casuali che tutti i PET normalmente mostrano per qualche istante al momen-





alla scansione della tastiera è montata al contrario. Su macchine dichiarate guaste è buona norma non dare per scontato che i pezzi siano tutti al posto giusto! Rimesso a posto l'integrato si rileva come la CPU riesca ora all'accensione a programmare le uscite della PIA, anche se questo non basta per far funzionare il computer.

to dell'avvio.

Rimessi tutti chip il computer non riesce ancora avviarsi e sul video appare una sequenza di parentesi, barre e trattini. Si riparte quindi con l'esame del processore e dei buffer (gli integrati 74LS244) analizzando con l'oscilloscopio i segnali sugli 8 terminali dei dati e sui 16 degli indirizzi. Un'intensa attività testimonia l'esecuzione del codice in ROM da parte della CPU. Ogni tanto però "la giostra" si ferma e non riparte nemmeno con la riaccensione. Si decide di sostituire il NE555 che controlla il reset.

Sul bus degli indirizzi si nota la mancanza di A5. Si scopre che una pista del circuito fa contatto con il dado metallico di fissaggio della scheda al contenitore. In origine doveva esserci una rondella isolante.

Si testa l'integrato 74154 che ha l'importante compito di selezionare (una alla volta) le periferiche interne al computer. Con la scheda privata degli integrati, è possibile verificare il buon funzionamento del 74154 mettendo a massa con collegamenti volanti i terminali A12-A13-A14-A15 del bus indirizzi. Nel rimettere a posto gli integrati ecco un'altra sorpresa: la PIA 6520 che sovrintende

### **Un problema di memorie.**

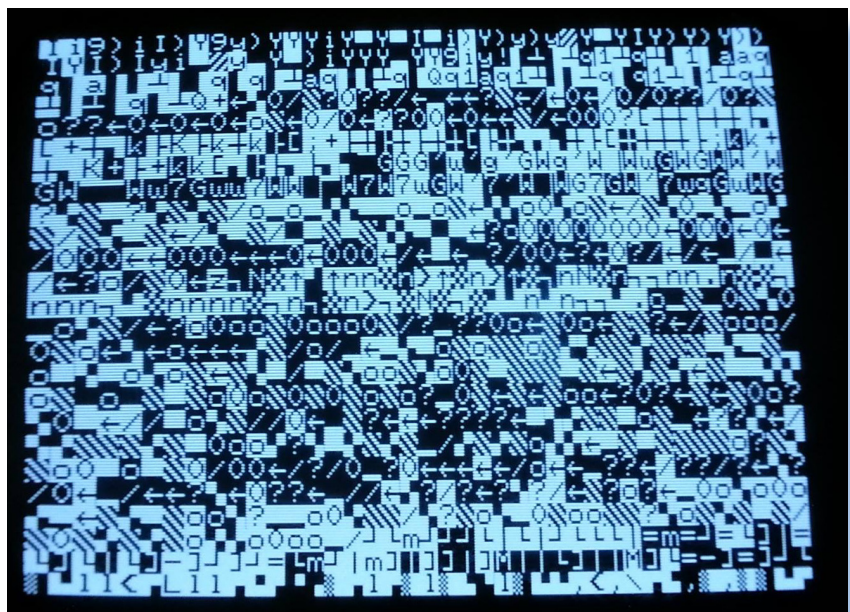
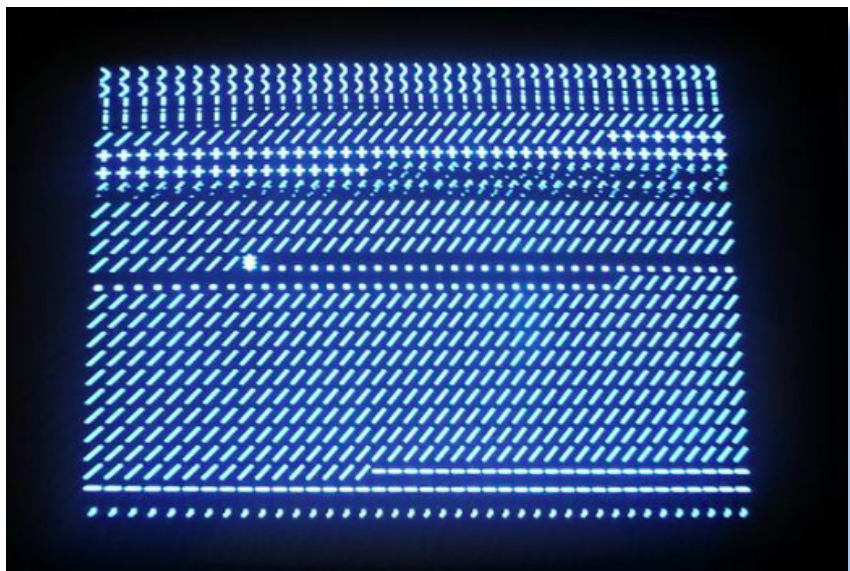
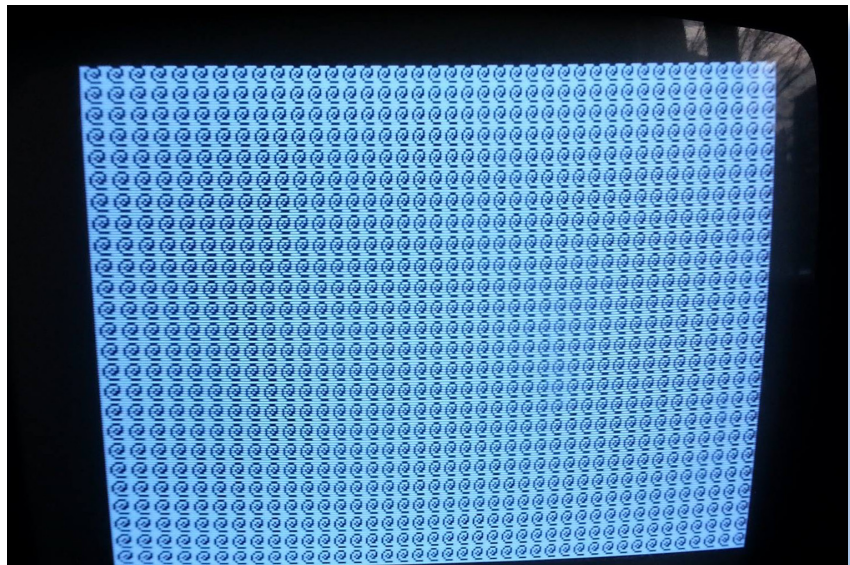
Resta il dubbio sul funzionamento delle RAM 6550 che equipaggiano sia il sistema sia la memoria video. Si tratta di integrati



con capacità di 1KB x 4 (ossia ce ne vogliono due per completare ogni banco da 1KB e 8 bit). I PET della prima serie, come questo, utilizzano 2 chip 6550 per la memoria video e ben 16 chip per dare al sistema la fantasmagorica dotazione di 8KB! Questi chip RAM non hanno una buona fama e sono stati prodotti per breve tempo, quasi solo per l'impiego su sistemi Commodore. Va da sé che siano introvabili, cosa che rende difficile disporre di qualche chip sicuramente funzionante per le prove. Come si fa a trovare una mela marcia tra 18 mele che sono tutte un po' vecchie e ammaccate?

Con alcuni tipi di memorie basta un dito per riconoscere i chip guasti perché scaldano in modo differente dagli altri, ma qui tutti i chip riscaldano molto. Chi ci aiuta?

Su Web spunta la pagina di *Matthew D'Asaro* che contiene due suggerimenti utili. Il primo riguarda la possibilità di avviare il PET con un ridotto numero di memorie: 4 chip (per 2K totali) popolando i 4 zoccoli a destra della scheda. Il secondo riguarda una semplice tecnica per capire se le RAM funzionano,



esaminandole “a vista” negli zoccoli del circuito video. In pratica, valutandone il funzionamento osservando sul monitor lo schema casuale di caratteri che appare all’avvio del PET. E dato che il sistema usa un solo tipo di memorie è solo questione di pazienza passarle due alla volta negli zoccoli video per fare il test.

Gli esempi sul sito fanno comprendere come le memorie 6550 già montate come video-RAM non siano funzionanti. Sostituite con due scelte a caso tra quelle del sistema, il “pattern” su schermo diventa davvero casuale, come prescritto. All’accensione la situazione è cambiata, sono sparite parentesi e barrette, lo schermo si cancella completamente e non appare nulla. Dopo tanto lavoro rieccoci davanti a uno schermo nero.

L’ipotesi più plausibile è che qualcuna delle quattro memorie scelte per equipaggiare il banco “minimo” da 2K non funzioni a dovere. Cambiando l’assortimento delle RAM con quelle nella vaschetta antistatica ecco l’attesa sorpresa: il vecchio PET classe 1978 torna a visualizzare la scritta d’avvio del Basic, sia pure con la disponibilità di soli 1023 byte di memoria.



### **Il test del sistema.**

Cosa c’entrano una volpe bruna e un cane pigro con un vecchio computer da provare? E’ per via della frase inglese “The quick brown fox jumps over the lazy dog” che contiene tutte le lettere dell’alfabeto ed è quindi perfetta per accertarsi del funzionamento della tastiera. Digitati anche i numeri da 0 a 9 abbiamo la conferma che la vecchia “chicklet” è in ottima forma, malgrado non sia il massimo come feedback. Cosa curiosa (ma normale) permette di scrivere solo in maiuscolo; i caratteri minuscoli, presenti in ROM, sono accessibili solo dai programmi. Poiché la tastiera è “letta” come matrice righe/colonne dalla PIA 6520 non c’è dubbio che tastiera, cavi e circuiti siano funzionanti. E’ stato utile, nella fase iniziale della pulizia, dissodare accuratamente i connettori.

Per evitare problemi d’avvio avevamo installato il sistema con il minimo sindacale in fatto di RAM (2KB). Si procede quindi a rimettere altre memorie RAM, un poco alla volta, facendo caso agli eventuali malfunzionamenti. Con altri 4 chip (2K di RAM) la disponibilità al prompt del Basic sale a 3071 byte. Con altre coppie di memorie si ottengono 4095 e poi 5119 byte. Le rimanenti memorie sono tutte difettose: alcune non aumentano la capacità, altre provocano il blocco del sistema.

Si prova ora a salvare e caricare programmi. Il motore del registratore incorporato nel PET viene correttamente acceso dal comando SAVE, così come con il comando LOAD, ma le rotelle non girano e il caricamento non riesce.

## **C'era una volta il nastro.**

Il registratore dati C2N incorporato nel PET 2001 non è molto diverso da un comune lettore per compact-cassette e soffre degli stessi problemi: dopo anni la cinghia si è allentata e non permette il trascinarsi del nastro. Per fortuna il reperimento di un ricambio e la sostituzione non sono un problema. Quest'ultima va preceduta dalla pulizia con alcool delle sedi della cinghia sulle pulegge per eliminare sporco o grasso che la farebbe scivolare. L'apertura del contenitore del registratore rivela la necessità di rinforzare con colla alcuni supporti delle viti e quindi di ripulire l'interno. L'esame del circuito evidenzia un collegamento di massa interrotto. Si sostituisce anche un condensatore da 22 $\mu$ F che appare in cattivo stato.

L'ulteriore prova di caricamento non riesce, anche se l'oscilloscopio rivela la presenza di onde quadre sul terminale READ. Il segnale READ si connette con il pin 40 della PIA che gestisce la tastiera. Questa PIA è la stessa che in precedenza abbiamo scoperto montata al contrario. La porta al pin 40 "al contrario" è in contatto con il terminale 20 dello zoccolo, connesso con i 5V dell'alimentazione. E' probabile quindi che abbia subito danni. Sostituita la PIA il caricamento dati riesce perfettamente.



## **Il PET ritrova la memoria.**

Con le memorie "buone" a bordo il PET ha ora 6KB di RAM e 5119 byte visibili dal Basic. Come fare per rimpiazzare 4 chip RAM mancanti e ripristinare il sistema com'era in origine? Purtroppo le vecchie RAM 6550 sono introvabili. Il sito di Matthew D'Asaro, già citato, propone in vendita alcuni adattatori che consentono di rimpiazzare memorie 6550 con le più comuni 2114. La ricerca su Web rivela anche l'esistenza di una scheda adattatrice da 8KB e che consente di rimpiazzare in un colpo solo l'intero banco delle 16 memorie 6550. Una versione dell'adattatore è stata realizzata in Italia dal mitico XAD Nightfall, al secolo Davide, grande esperto nel recupero dei retro computer che, contattato, ci invia lo stampato dell'adattatore.

La scheda dev'essere completata con una memoria statica (6264 in configurazione 8K x 8bit) e un integrato TTL (74LS00) che simula il sistema di abilitazione delle 6550.

E' progettata per prelevare alimentazione, bus e segnali direttamente dagli zoccoli vuoti della RAM del PET. Per funzionare correttamente in sostituzione di tutte le RAM dev'essere montata a cavallo tra gli zoccoli 4 e 6 in modo da poter prelevare i due segnali di selezione (SELO e SEL1) con cui la CPU abilita alternativamente il primo o il secondo banco di memoria da 4KB. Poiché nel nostro sistema c'è già un banco RAM da 4KB già funzionante perché rinunciare a usarlo? La soluzione è in una semplice modifica per abilitare la scheda non sul primo e secondo blocco da 4KB, ma sul secondo e terzo, aggiungendo quindi gli 8KB della nuova memoria ai 4KB esistenti, per raggiungere la capacità totale di 12KB. Non è possibile sfruttare tutte e 12 le memorie 6550 "buone" del sistema (6KB) perché la decodifica degli indirizzi ha solo 16 uscite e lavora per blocchi da 4KB (16 x 4K = 64K). Questo non permetterebbe di cucire insieme la mappa di memoria senza lasciare "un buco" da 2KB nel secondo banco. Per funzionare nel modo previsto, la scheda



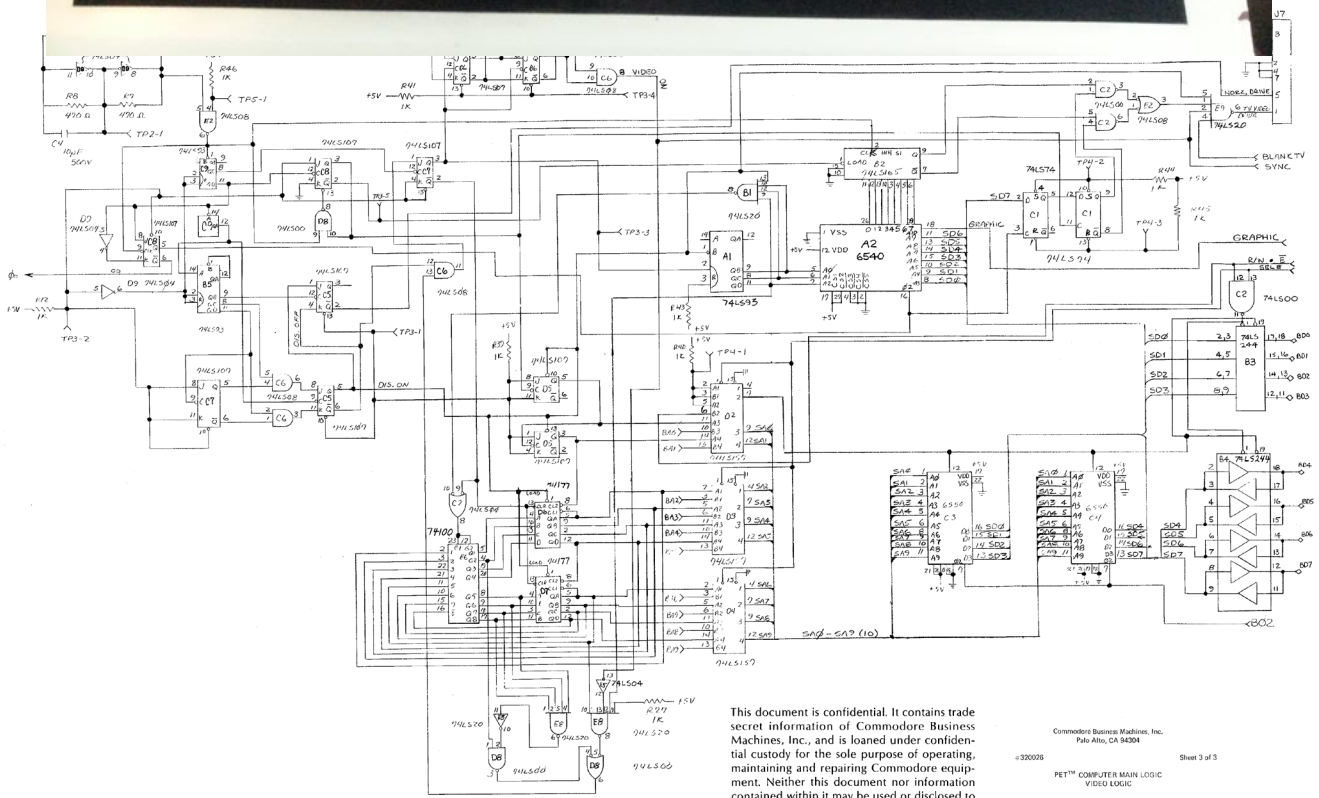
adattatrice va montata a cavallo degli zoccoli 5-7 delle RAM e modificata in modo da prelevare con un filo l'ingresso di selezione SELO per poterlo collegare con il SEL2 della piastra madre, corrispondente al terzo banco da 4KB. Questo segnale, presente sul pettine d'espansione è prelevabile con maggiore comodità sul pin n.3 dell'integrato 74154.

Per garantire il corretto il funzionamento dei buffer sulle linee dati della RAM fino a 12KB occorre fare una piccola modifica sulla piastra madre che consiste nel rimuovere il cavallotto "8K" a fianco dell'integrato 74154 e al suo posto montare un diodo per piccoli segnali (tipo 1N4148) con il positivo rivolto verso la pista che conduce al pin 2 dell'integrato 74154. Un altro diodo è saldato tra il terminale negativo del primo diodo e SEL2 (pin 3). Con questa modifica, facilmente reversibile, il PET può funzionare a 12KB di cui circa 11 visibili dal Basic. E qui viene il bello. Provare a riscoprire cosa si può fare con un computer da soli 12KB.

(\*)



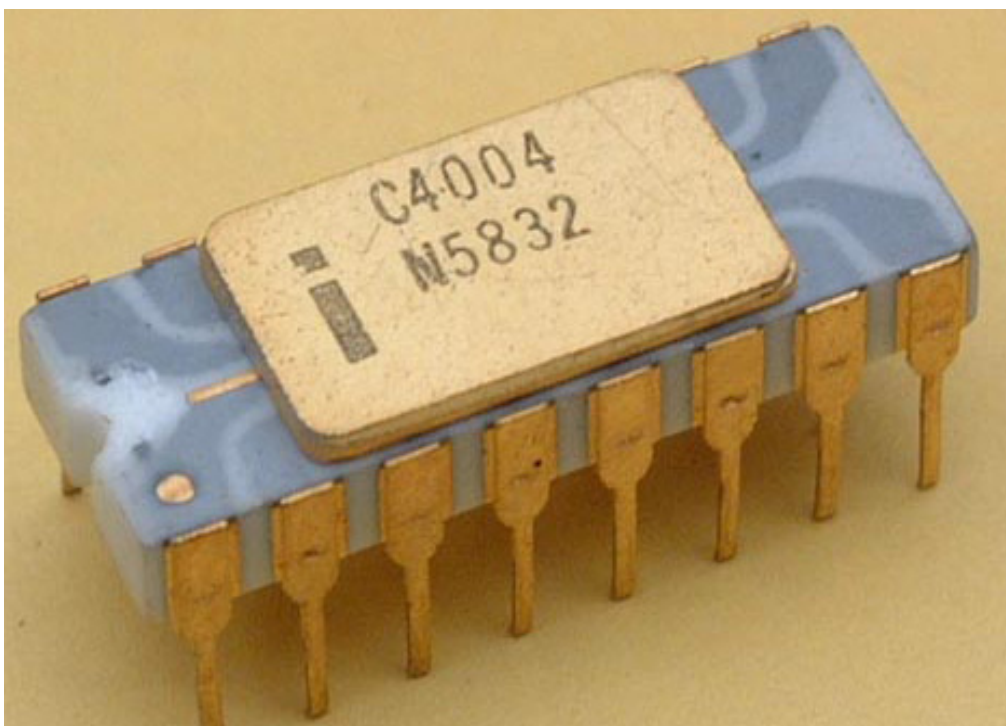




This document is confidential. It contains trade secret information of Commodore Business Machines, Inc., and is loaned under confidential custody for the sole purpose of operating, maintaining and repairing Commodore equipment. Neither this document nor information contained within it may be used or disclosed to anyone, or reproduced for other purposes without the written consent of Commodore Business Machines, Inc.

Commodore Business Machines, Inc.  
Palo Alto, CA 94304  
© 2002026  
PET™ COMPUTER MAIN LOGIC VIDEO LOGIC  
pertaining to board assembly - 320026  
NOTE: All markings are under copyright of Commodore Business Machines, Inc. All rights reserved. 320132, 320081, 320132, 320081.  
The identifying board number is on upper left corner.

## La nascita dell'INTEL 4004



di **Tullio Nicolussi**

Negli anni '50 i computer utilizzavano un misto di tecnologie elettromeccaniche (relè) ed elettroniche come tubi a vuoto e in minima parte transistor. Era ben evidente che l'evoluzione avrebbe portato l'abbandono delle valvole termoioniche in favore di elementi a stato solido per ovvie ragioni di ingombro, consumo e affidabilità. La cosa avvenne con una certa gradualità e fin dai primi anni '60 si cominciarono a vedere i primi circuiti integrati con una scala di integrazione tale da permettere l'inserimento in un singolo chip di una dozzina di transistor.

Da quel momento si innescò un circolo virtuoso che portò al raddoppio della scala di integrazione ogni anno, cosicché già alla metà del decennio si era in grado di sostituire gli elementi magnetici che costituivano la memoria dei computer esistenti con circuiti integrati.

Federico Faggin, uno dei protagonisti della rivoluzione chiamata microprocessore, ricorda i suoi primi anni alla Intel e l'idea di riuscire a costruire un computer su singolo chip. "Alla fine degli anni '60" - egli dice

- "era già possibile, solo che non era conveniente dal punto di vista economico."

Ted Hoff, un'altro protagonista di quella corsa tecnologica e ingegnere alla Intel, afferma che non solo Intel lavorava all'idea del computer su un chip, ma anche IBM, RCA, SRI erano interessate in prima persona mentre altre aziende contribuivano all'evoluzione. Fra queste si ricorda Fairchild, Rockwell, General Instruments e Texas Instruments. Analogamente in Giappone la Sharp Corporation stava discutendo su idee simili.

Lo sviluppo del "computer su un chip" non era la via principale sulla strada dell'evoluzione elettronica. Sempre di più si costruivano chip con complessità crescente per venire incontro alle esigenze di automazione di vari settori. Alla fine degli anni '60 - ricorda Robert Noyce, allora presidente di Intel - la complessità dei circuiti era tale che i costi di progettazione stavano esplodendo. Lo sviluppo di un sistema programmabile era assolutamente vitale per l'industria elettronica.

*Ma andiamo con ordine e scopriamo cosa stava facendo il Giappone, allora unico paese industrializzato che faceva da contraltare all'industria elettronica statunitense.*

*Lo sviluppo dell'industria elettronica giapponese stava seguendo una strada pressoché parallela a quella degli Stati Uniti. Il nome chiave da ricordare è un certo ingegnere Tadashi Sasaki.*

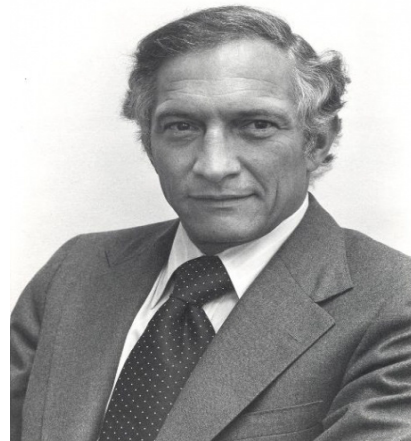
*Partendo dai lavori pionieristici di Tadashi Sasaki che approdò alla Fujitsu nel 1963, gli ingegneri e fisici giapponesi contribuiscono in maniera definitiva alle ricerche nel campo dei semiconduttori. La prova è che Leo Esaki, che aveva lavorato con Sasaki proprio in quegli anni, ricevette nel 1973 il premio Nobel per le sue ricerche sui semiconduttori.*

*Nel periodo di tempo in cui Sasaki lavorò presso Sharp, egli organizzò una squadra di giovani ingegneri elettronici che seguendo le sue indicazioni lavorarono all'idea del chip ad alta integrazione. La Sharp è stata la prima industria mondiale a produrre un calcolatore interamente elettronico nel 1964.*

*Gli storici calcolano che la Sharp fosse avanti di due anni rispetto all'industria*



Tadashi Sasaki



Robert Noyce

*americana nell'ingegnerizzazione dei calcolatori e gli americani "buttavano un occhio" al di là del Pacifico...*

*Sasaki conosceva bene le ricerche dei suoi colleghi americani dei quali godeva molto stima e indirizzò le ricerche non verso l'integrazione sempre più spinta di componenti ma verso l'evoluzione delle funzionalità dei chip, cioè sulla strada che portava ai microprocessori.*

*Per ragioni post belliche l'industria giapponese non poteva produrre in proprio i componenti elettronici integrati, cosicché ad un certo punto alla Rockwell arrivò una strana richiesta dal suo cliente Sharp a firma di Sasaki: produrre una famiglia di quattro chip con funzionalità complementari. Sasaki aveva elaborato l'idea teorica di questo set di componenti ma mancava la realizzazione pratica che la Rockwell avrebbe potuto realizzare dopo che gli stessi giapponesi avessero messo a punto i dettagli elettronici dei chip. La Rockwell rifiutò giudicando la cosa poco interessante per il proprio business che consisteva nel vendere componenti elementari anche alla Sharp stessa.*

*Quanto ebbe poi occasione di pentirsene!*

*Arriviamo al 1968 e ad una data molto importante per lo sviluppo dell'informa-*

Ted Hoff



tica moderna: il 18 luglio 1968 quando a Mountain View in California viene fondata la Intel Corporation. Risulta che pochi mesi dopo il suo presidente Robert Noyce andò in Giappone e incontrò Sasaki

nel suo ufficio alla Sharp. L'intento di Noyce era vendere chip alla Sharp come fornitore in concorrenza alla Rockwell ma l'azienda giapponese aveva un contratto di fornitura in esclusiva con la Rockwell. Sasaki stava cercando una azienda lo seguisse sulla strada dei chip programmabili e la Intel, appena fondata e affamata di commesse era l'interlocutore ideale.

La Rockwell si rifiutò di aprire alla Intel la possibilità di diventare fornitore della Sharp e a questo punto Sasaki decise di fare da solo coinvolgendo un imprenditore giapponese che aveva costruito una piccola ma interessante industria elettronica chiamata Busicom: un certo Yoshio Kojima che conosceva personalmente Sasaki e che aveva da quest'ultimo dei consigli sullo sviluppo del loro piccolo business, accettò lo strano accordo per cui sarebbe stato lui a chiedere ad Intel di sviluppare i chip dell'innovativo progetto di Sasaki.

L'ingegnere a capo del progetto per la Busicom fu Masatoshi Shima che si prese l'onere di convertire le idee di Sasaki in schemi elettrici e lavorare a stretto contatto con il fornitore Intel per l'ingegnerizzazione del progetto, cioè per la mascheratura dei chip e tutti i dettagli produttivi.

Alla Intel fu l'ingegnere Ted Hoff che dove-

va realizzare materialmente i chip con le mascherature e tutta la parte tecnologica necessaria. Egli si convinse che i chip progettati da Shima potevano essere migliorati soprattutto per farli rientrare nei fattori di forma che erano standard per l'industria. Hoff propose i quattro chip che erano siglati: 4001 - 2048 Bit ROM; 4002 - 320 bit di RAM; 4003 - uno shift register e porta seriale e parallela; 4004 - la CPU a 4 bit.

In realtà non era proprio il progetto originale definito con la Busicom che accettò con riluttanza che il progetto fosse cambiato ma alla fine si convinse. Era la fine del 1969 e i progettisti giapponesi tornarono negli States in Aprile dell'anno dopo per scoprire che il progetto non era più diretto da Hoff, intanto deviato sul più complesso progetto 8008, ma da un nuovo ingegnere assunto appena una settimana prima: un certo Federico Faggin.

Ecco dove si inserisce il "genio italiano", in un progetto già avviato e che abbisogna certo di competenze specifiche e capacità di project management prima di tutto. Peraltro Faggin non si è mai dichiarato inventore del microprocessore ma piuttosto un creatore di soluzioni, lavoro da vero ingegnere!

I responsabili della Busicom sollevarono dubbi sulla nuova conduzione del progetto ma non potevano che accettare. Faggin si



Masatoshi Shima

Federico Faggin



rivelò la scelta giusta in particolare per la realizzazione del chip più complesso: la CPU 4004 con i suoi oltre 2000 transistor.

Pochi mesi dopo la consegna del lavoro da parte di Intel nel 1971, la Basicom fu in grado di commercializzare una calcolatrice da tavolo che usava il nuovo chip set e in particolare il microprocessore 4004.

### Conclusione.

Le storie che si intrecciano relativamente alle circostanze che hanno permesso la nascita del primo microprocessore sono molteplici, come molteplici sono i punti di vista.

Come spesso succede per le realizzazioni di successo, molti se ne attribuiscono i meriti o quantomeno la scoperta di singoli passaggi tecnici indispensabili. Così come molti, anche "addetti ai lavori" semplificano eccessivamente o prendono delle cantonate, come quella di attribuire a Faggin la "scoperta" del microprocessore se non addirittura dello stesso transistor (trasmissione di Radio3, "Radiotre Scienza" del 5/01/2015). davvero una "svista" (speriamo sia stato solo un refuso) abbastanza inquietante, anche perché l'affermazione è di un docente universitario!

### Riferimenti:

- William Aspray; "The Intel 4004 Microprocessor: What Constituted Invention?"; su *IEEE Annals of the History of Computing*, Vol. 19, No. 3, 1997;

- *Global History Network*: <http://www.ieeeahn.org/wiki/index.php/Special:Home>;
- *Sharp history*: [http://www.sharp-world.com/corporate/info/his/h\\_company/1964\\_1966/index.html](http://www.sharp-world.com/corporate/info/his/h_company/1964_1966/index.html);
- *Podcast della trasmissione Radio3 Scienza*: <http://www.radio.rai.it/radio3/podcast/rssradio3.jsp?id=273>;
- *Federico Faggin*; "The Making of First Microprocessor"; *IEEE SOLID-STATE CIRCUITS MAGAZINE*, Winter 2009, DOI: 10.1109/MSSC.2008.930938;
- *F. Faggin, M. Shima, M.E. Hoff, H. Feeney, S. Mazor*; "The MCS-4 - An LSI micro Computer System"; in *Proceedings of the IEEE '72 Region Six Conference*; DOI: 10.1109/MSSC.2009.931980

(\*)



Basicom 141 P - la prima serie di macchine calcolatrici ad usare il mp 4004.



o meno palesemente con le nostre risultanze personali. Ricordo un gioco per PC degli anni '90, "Prey" si chiamava, che doveva essere l'ultimate della simulazione di volo da combattimento: una ciofecca indicibile! Eppure la testata che leggevo io gli dava 9/10... non ho mai capito perché e non mi ricordo nemmeno molto di questo gioco, anche perché sono sempre stato un giocatore "sporadico".

Nella letteratura scientifica c'è un mantra che viene ripetuto da coloro che cercano l'affidabilità: "referee". Il "referaggio", detto in brutto italiano, sarebbe una specie di garanzia di affidabilità. Un articolo referato (usiamo questo brutto vocabolo) si assume sia stato esaminato ed approvato da "cultori della materia", cioè persone che di quel tema se ne intendono e quindi possono esprimere un giudizio di merito obiettivo.

Per ragioni di sicurezza non si sa chi siano i referee contattati dall'editore. Questo per evitare le consuete furbizie di parte dell'accademia (italiana in primis) che più che dedicarsi alla ricerca sembra dedita a cercare le scorciatoie per emergere dal mare della mediocrità.

Ma dove non possono gli autori ecco che possono gli editori: disonesti ovviamente, parliamo di quella minoranza.

C'è una vicenda alquanto divertente che vi voglio raccontare.

C'è una pubblicazione che si chiama "International Journal of Advanced Computer Technology (IJACT)", notate l'enfasi nel nome, con tanto di ISSN 2319-7900, quindi una iniziativa registrata presso gli appositi enti che ospita articoli di tecnologia riguardanti l'informatica.

Bene, un certo Peter Vamplew, un docente australiano

che - stanco di ricevere spam - ha inviato un articolo civetta alla rivista. L'articolo ha come titolo "Get me off Your Fucking Mailing List" e si presenta con il layout tipico degli articoli scientifici con tanto di grafici e formule, solo che il testo è una ripetizione della stesa frase del titolo per tutte le dieci pagine!

L'articolo si può scaricare a questo indirizzo: <http://www.scs.stanford.edu/~dm/home/papers/remove.pdf>. Il caso ha fatto notizia perché l'articolo, dietro il pagamento di 150 dollari USA, è stato accettato per la pubblicazione con tanto di scheda valutativa dei presunti referee (a questo indirizzo la scheda originale: <https://scholarlyoa.files.wordpress.com/2014/11/review-form.pdf>) che ne avalla la scientificità e lo giudica addirittura "Excellent" in certi aspetti.

Una vicenda divertente che svela la poca serietà dell'editore e quindi della pubblicazione. Anche in Internet l'abito (il nome) non fa il monaco.

(\*)

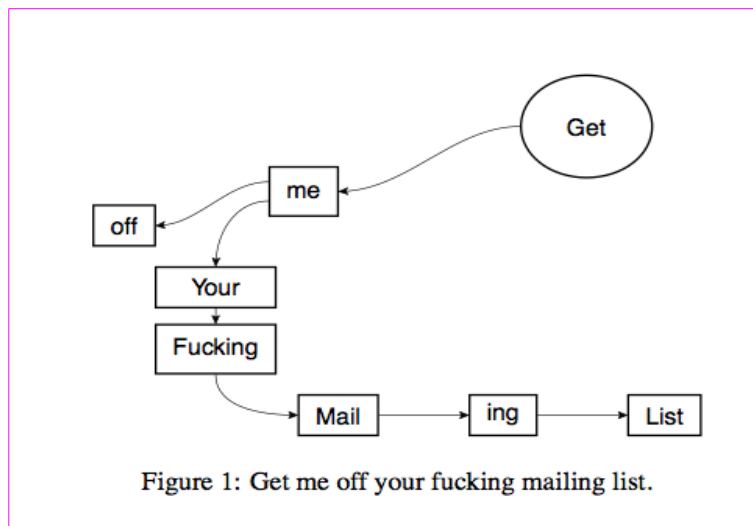


Figure 1: Get me off your fucking mailing list.

# Francomputer Club Texas



*Di Sonicher*

## Scheda

*Titolo: Tutto TI 99*

*Autore: Vari*

*Editore:*

*Francomputer Club Texas*

*Anno: 1982*

*Lingua: Italiano*

*Pagine: 50 (circa)*

*Costo: L. 4.000 a copia*

Le macchine di calcolo home della Texas e in particolare il TI-99/A, straordinario calcolatore personale, non hanno goduto di un supporto editoriale curato dalla stessa azienda Texas Instruments. Questo è stato un gran peccato perché il sistema meritava sicuramente migliore fortuna e non solo in Italia! E migliore fortuna meritavano gli acquirenti che a suo tempo si sono fatti allettare dalle strepitose caratteristiche del TI-99/A, il primo 16 bit alla portata dell'hobbista mai apparso sul mercato!

Le politiche commerciali hanno deciso diversamente e ben presto la casa madre ha rinnegato la sua creatura facendole mancare quel supporto e quelle prospettive di sviluppo che sono stati il vero motore dell'introduzione del personal computer nel mondo.

A questa mancanza di un "punto di accumulazione" hanno supplito per quello che era possibile le riviste del settore che riservavano più o meno regolarmente un po' di spazio ai listati e alle curiosità tecniche relative al sistema.



Da m&p computer a MC microcomputer a Bit a Personal Software a Papersoft e qualche altra pubblicazione, tutte hanno "tenuto botta" per un po', convinte loro stesse che la base di installato in Italia valesse la pena. Ovviamente la cosa non poteva durare a lungo e soprattutto in un'epoca dove le novità e gli abbandoni si susseguivano a ritmo incalzante. Così anche i listati per TI-99/A hanno cominciato a scarseggiare e all'epoca purtroppo non c'era Internet e la reperibilità delle riviste anglosassoni dedicate alla macchina, era problematica.

Per assurdo chi prende oggi un TI99-4/A ha a sua disposizione molto più materiale di chi l'ha comprato al momento del lancio!

Una fonte utile di informazione sono stati in questo caso i bollettini dei club che spontaneamente si formavano attorno all'interesse per una particolare macchina. Numerosissimi i club per Spectrum e C64, ma anche i sistemi meno diffusi hanno avuto le loro aggregazioni.

Francomputer Club Texas è una di queste associazioni. Guidata da un vulcanico piccolo imprenditore di Vicenza, Fanco Gonzato, si è fatta carico di divulgare la conoscenza della macchina guadagnandoci qualche spicciolo sia con l'invio della fanzine cartacea che con la vendita dei programmi che offriva a prezzi davvero popolari (dalle 5.000 Lire in su) oltre che espansioni hardware, cartucce etc...

Oltre alla macchina Texas, Francomputer è noto anche per un'altro sistema che in Italia ha avuto poca fortuna: il Sega 3000 del quale credo che Gonzato sia stato importatore non esclusivo.

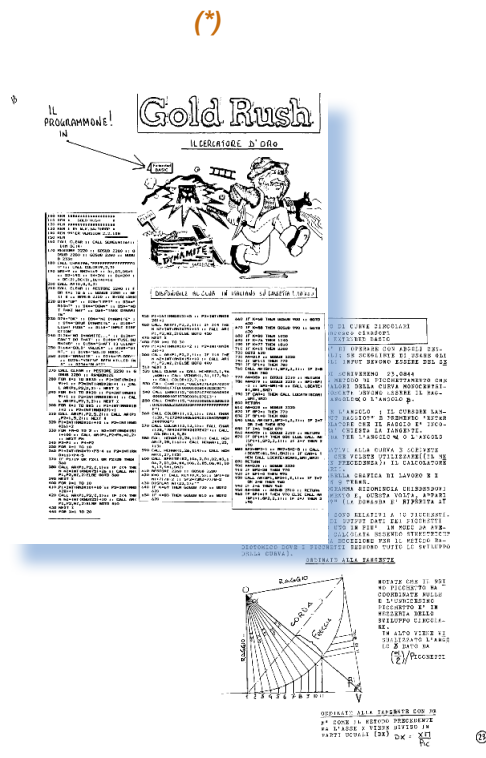
Il signor Gonzato ha donato i primi due numeri al gruppo "TI-99 Italian User Group" ([www.ti99uc.it](http://www.ti99uc.it)) che li ha scansionati e resi disponibili a tutti gli appassionati sia della macchina che di storia del calcolo personale.

La pubblicazione non è spartana, è "ultra-spartana"! Realizzata con macchina da

scrivere/stampante ad aghi, con disegni e annotazioni a mano è impaginata alla meglio, non supererebbe oggi l'esame di nessun utilizzatore, ma all'epoca era sicuramente ciò che viene definito "grasso che cola".

Spesso è più importante per l'hobbista sapere di non essere solo piuttosto che qualsivoglia altra caratteristica del proprio sistema. Non essere soli vuol dire condividere (non solo i programmi) e quindi di fatto accrescere le proprie conoscenze.

Se posso azzardare una ipotesi probabilmente i programmatori e tecnici informatici migliori sono venuti fuori proprio da chi non aveva il solito C64. Proprio perché dovendo sporcarsi le mani si è creato quella mentalità che oggi chiameremo "da nerd" più di coloro che i programmi li trovavano già belle e pronti sulle cassette vendute in edicola.



## Western-Digital WD-90



di Tullio Nicolussi

*Nel primo decennio di utilizzo dei microprocessori come cuore di sistemi di calcolo a basso costo (approssimativamente fra il 1975 e il 1985) e prima che si imponessero gli standard di fatto, grazie ai grandi volumi di vendita, dei vari 8080, Z80 e 6502, si assistette alla nascita di alcune iniziative "esoteriche".*

*Partendo dai lavori pionieristici della Intel sui processori 4004 e 8008, altre aziende presero la mossa per progettare i propri micro, cercando ovviamente di migliorarne le prestazioni e l'usabilità.*

*Uno dei campi di miglioramento era l'ingegnerizzazione dei linguaggi di programmazione ad alto livello. Usare l'assembler era normale in ambito tecnico ma certo non si prestava a realizzare quelle applicazioni che si stavano affacciando nel mondo dell'elaborazione personale. Parliamo ad esempio del processamento dei testi e della grafica.*

*Purtroppo la traduzione da codice sor-*

*gente a codice oggetto, il lavoro cioè appannaggio dai compilatori, era fonte di inefficienze nell'ottimizzazione. Venivano (e vengono introdotti tutt'oggi) molti artefatti fra la logica del sorgente ad alto livello e il codice tradotto in linguaggio macchina, che si traducono poi in un assemblato più lungo e meno veloce. La velocità non era forse ancora un ostacolo percepito ma l'occupazione di memoria lo era sicuramente! Macchine che avevano pochi Kbyte di memoria RAM erano sensibili eccome allo spazio occupato dai programmi, per non parlare della limitatissima capacità di indirizzamento delle prime CPU!*

*I linguaggi interpretati non potevano essere la risposta e lo si sapeva; fino alla metà degli anni '80 resistette il BASIC, grazie al suo utilizzo massiccio nelle macchine home, ma poi anche questo cedette il passo al compilatore.*

## **Contesto storico.**

*Già nel 1975 si era pensato ad una terza via: perché non costruire un microprocessore in grado di eseguire direttamente il codice ad alto livello? La strada sembrava percorribile e si cominciò con il LISP. Il LISP è un linguaggio dalla struttura matematica molto spiccata essendo basato sulla cosiddetta lambda-notation che assomiglia molto alla gestione di uno stack. Le istruzioni scritte si possono facilmente mettere in relazione con operazioni elementari, campo appunto dei processori. Le LISP Machine sono state delle meteore nella storia dell'elaborazione anche perché, tolto il campo della ricerca informatica in direzione dell'intelligenza artificiale, il LISP non si è diffuso in maniera trasversale.*

*Stessa sorte capiterà qualche anno più tardi al Prolog, altro linguaggio "logico" che ha fornito spunto per la costruzione di processori ad-hoc.*

*Il Forth, all'inizio degli anni '80, ha avuto la sua chance. Il Forth è un linguaggio stack-oriented con il quale il programmatore si trova ad interagire con una macchina virtuale abbastanza elementare. Tradurre un listato Forth in pseudo-codice macchina è abbastanza banale, tanto è vero che rispetto ai "fratelli" il Forth è velocissimo! E' stato spremuto ancora di più con la progettazione di processori specializzati.*

*Queste prime soluzioni: LISP, Prolog e Forth hanno una cosa in comune: sono ostici da imparare e poco documentativi mentre per diffondere la programmazione bisognava andare nella direzione di un linguaggio più "umano".*

*Il Pascal come linguaggio di programmazione ha avuto i suoi momenti di gloria ponendosi come killer-application nei confronti del BASIC, tacciato di in-strutturalità conclamata. Aveva anche*

*un'altra freccia al suo arco: era teorizzato molto puntualmente da Niklaus Wirth, come si sa.*

*La sua implementazione pratica più famosa e diffusa è il USCD Pascal (di proprietà dell'Università di San Diego in California) che traduce il sorgente Pascal in P-code (una sorta di pre-assemblato) che viene poi eseguito da una macchina virtuale. E' il paradigma che poi è stato cavalcato da Java (partendo dal 1996 circa) e ai nostri giorni dai linguaggi del framework .NET di Microsoft e suoi cloni/derivati.*

*Cercare di costruire una macchina virtuale per il P-code che sia la più efficiente possibile, porta alla realizzazione di un chip specializzato. Così nascono le Pascal-Machine che per qualche anno mantengono effettivamente la promessa di una elaborazione superiore di circa un fattore 10 (sono dieci volte più veloci) rispetto a quella ottenibile con processori general-purpose.*

*I microprocessori tradizionali si riprenderanno presto la leadership prestazionale grazie alla loro diffusione e quindi alla disponibilità di fondi per la ricerca di nuove soluzioni e di un incremento prestazionale senza rivali.*

*Una Pascal-Machine la costruì nel 1979 la Western-Digital (che oggi commercializza praticamente solo hard disk) progettando*



e realizzando un suo set di chip specializzati conosciuti con la sigla WD-9000. Con una curiosa escalation inversa nelle sigle, la Western-Digital ha realizzato un calcolatore su scheda, il prodotto "Pascal MicroEngine WD-900", e poi una soluzione completa di interfacce e periferiche: il calcolatore WD-90.

Il computer su scheda costava 3.000 dollari mentre il sistema completo WD-90 arrivava a 5.000 dollari ai quali bisognava aggiungere un video-terminale e una/due unità floppy; alla fine una macchina completa si avvicinava molto ai 10.000 dollari.

Il prodotto rimase a listino e fu aggiornato fino al 1983.

### **Hardware.**

Stante la composizione del sistema si deve partire a considerare il chip-set che è deputato all'esecuzione del P-Code.

Il WD-9000 è composto da tre elementi costruiti dalla Western-Digital come evoluzione di un progetto precedente, siglato MCP-1600, usato fra l'altro nel PDP/11.

La tipologia di oggetti è la seguente:

CP2151 Data-ALU;

CP2161 Control;

CP1631 MICROM.

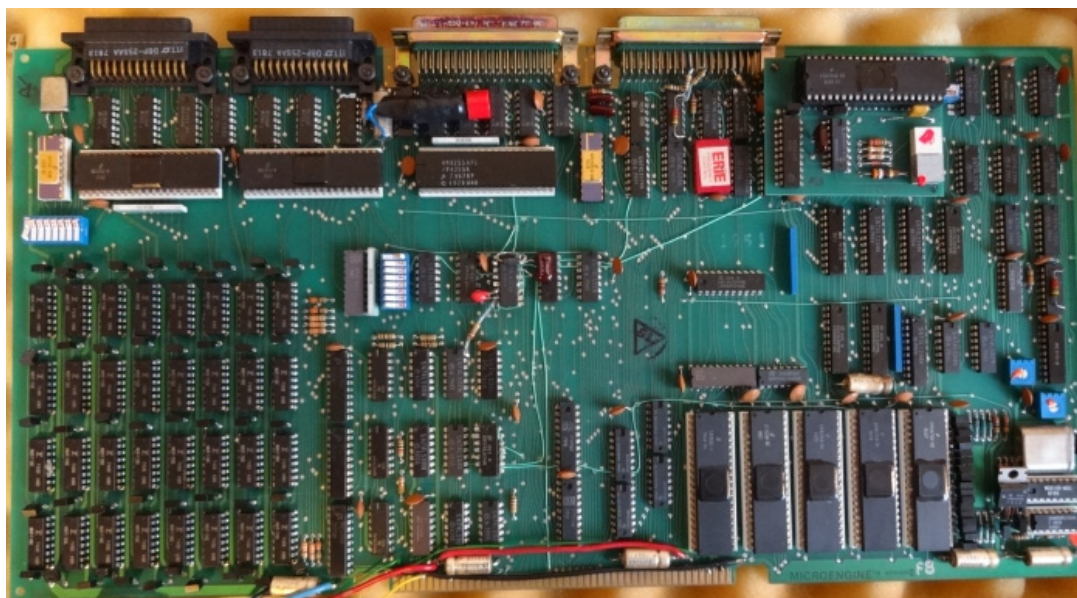
La parte di calcolo viene svolta dal chip CP2151 che contiene l'unità ALU e alcuni registri di dati; il CP2161 è un processore di stack (il P-code è un codice stack-oriented) ed infine il chip MICROM, che in pratica è una ROM, contiene il microcodice.

L'idea del micro codice in ROM permette una certa scalarità dei progetti e teoricamente l'adattamento ad altre versioni/linguaggi che producano P-code. Nel prodotto WD-900 vengono usati tre chip MICROM. Nel corso degli anni di commercializzazione la Micro-Engine (così venne chiamata da Western-Digital la società fondata ad-hoc per commercializzare il prodotto) diede corso a quattro aggiornamenti che richiedevano la sostituzione di uno fino a tutti e tre i chip MICROM.

L'ultima versione disponibile è aggiornata al UCSD III P-system.

Il chip-set va considerato nella sua interezza come costituente una CPU logica a 16 bit che funziona con clock a 3.3 MHz.

Il chip-set WD-9000, costruito con tecnolo-



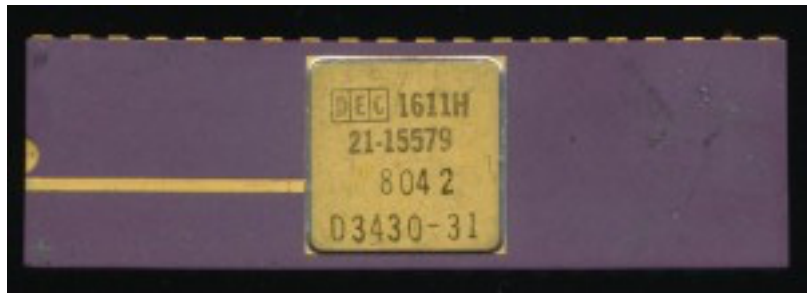
gia NMOS, richiede quattro alimentazioni: +/- 5V e +/- 12V e un clock a quattro fasi.

Su questo chip-set/CPU è costruita la scheda denominata WD-900, in pratica un micro-elaboratore a scheda singola che contiene anche un controller per floppy e per le periferiche di I/O, oltre ad una dotazione di base di memoria.

I possessori delle prime versioni della scheda WD-900 riportano lo sconcerto nel constatare le numerosissime correzioni "al volo", segno evidente di un progetto commercializzato troppo presto e forse anche di una certa difficoltà nell'ingegnerizzazione, causa questa forse di una morte prematura della soluzione WD per il mercato delle workstation P-Machine.

Sulla faccia-componenti della scheda si individuano subito i cinque chip a 40 piedini, ben allineati che costituiscono la CPU e il banco di RAM costituito da 32 chip, sulla sinistra. In alto in prossimità delle uscite i chip di controllo per l'I/O. Partendo da sinistra le due seriali, la porta parallela (doppia) e il connettore per le unità floppy.

Il computer WD-90 contiene questa scheda l'alimentatore e una espansione di RAM da 64 KByte (in realtà sono 32 Kbyte di celle a 16 bit). La RAM è dinamica e un circuito apposi-



to provvede al compito di refresh.

Acquistando un Wd-90 si deve poi disporre di un video terminale ed eventualmente di una/due unità floppy che sono supportate dalla scheda centrale ma che devono essere aggiunte all'esterno del cabinet. Nel 1979 i floppy drive erano tutt'altro che slim!

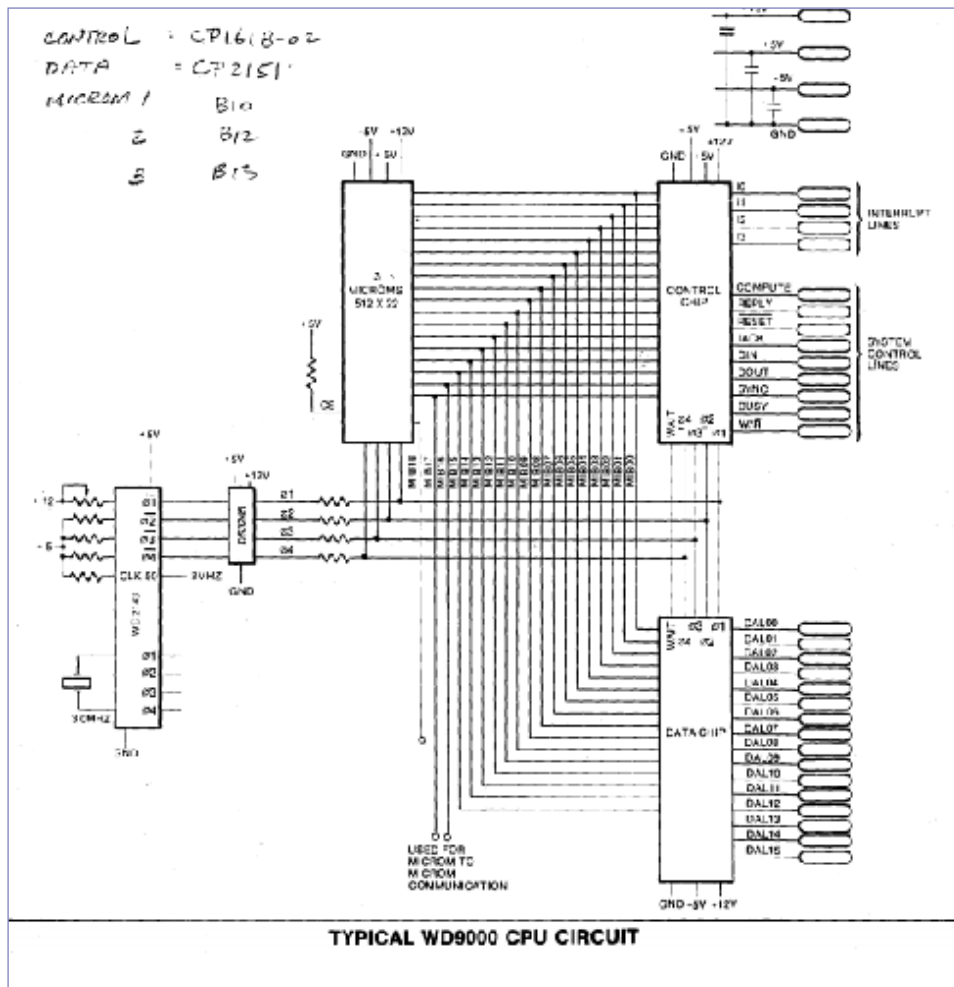
Oltre alla CPU la scheda contiene due porte seriali RS232 programmabili, due porte parallele a 8 bit e un controller per unità floppy da singola o doppia densità con fattore di forma sia da 5,25" che da 8".

Prima di operare il sistema va settato con tre blocchetti di dip-switch che si trovano sulla piastra. Va scelta la tipologia di floppy, la velocità delle seriali e la tipologia di terminale che si intende collegare. Il terminale andrà collegato alla porta seriale denominata A. I parametri impostabili sono molti e assolvono sicuramente al compito di collegare una vasta tipologia di terminali.

Sul retro emergono i quattro connettori più un bottone per il reset e lo switch dell'alimentatore.

La MicroEngine ha commercializzato anche una versione della scheda WD-900 su bus S-100 che era un bus industriale molto diffuso negli anni '70-'80. Fu anche curata la par-





te sviluppo con la disponibilità verso la fine della commercializzazione anche di un compilatore ADA, oltre che delle più recenti versioni del Pascal. La rete cita anche progetti "esoterici" con più schede WD-900 per la realizzazione di un computer multi-CPU. Si riportano anche soluzioni industriali "ibride" contenenti ad esempio una CPU Z80 dove il WD-9000 sarebbe usato come co-processore numerico.

### **Il Software.**

Il sistema operativo è una versione del UCSD Pascal Operating System che viene fornito su un supporto da 8" o su due da 5,25". Dopo l'accensione va premuto il tasto di reset per fare in modo che il sistema carichi l'UCSD Pascal dalla prima unità floppy.

Il sistema operativo consiste in un file manager, un text editor full-screen, un debugger e ovviamente il compilatore Pascal. Esiste anche un compilatore BASIC che evidentemente produce anch'esso P-code, altrimenti non sarebbe eseguito sulla macchina!

Per chi ha usato ancora l'UCSD l'operatività è semplice ed esattamente quella che si può trovare su un sistema home come l'Apple II.

La prima riga video è una riga di stato che riporta il mnemonico del comando con la prima lettera seguita da parentesi chiusa:

E)dit R)un F)ile C)omp....

seguono poi menù di secondo livello che spe-

cificano la tipologia di funzione da eseguire. Ad esempio per la scelta F)ile si può scegliere al secondo livello se salvare l'attuale area di lavoro oppure caricare un file o rinominarlo o cancellarlo,... Insomma le normali operazioni cui supervisiona un sistema operativo orientato alla gestione dei dischi.

Il funzionamento del sistema UCSD si basa sulla presenza di una work-area dove viene editato il sorgente, compilato, eseguito e debuggato. Tutti i comandi fanno riferimento all'area attiva che è unica.

L'editor del sistema UCSD Pascal è espressamente dedicato allo sviluppo del codice. Ci sono funzioni di controllo del testo (Find/Replace, etc...ma anche di "abbellimento" come l'indentazione automatica che rende il sorgente maggiormente leggibile (a patto di adattarsi ai concetti estetici predefiniti).

Per le installazioni che non utilizzano un videoterminale ma una telescrivente come unità di input/output, viene rilasciato un'altro editor, chiamato YALOE, orientato alla linea. Per chi non ha mai usato un sistema così configurato sembra strano e macchinoso dover interagire con il calcolatore con una specie di macchina da scrivere. Personalmente l'ho usato poco questo sistema ai tempi del PDP/11 dell'Università, ma ricordo che tutto sommato (a parte il fastidioso rumore) non era poi tanto male...

Il compilatore è a passaggio singolo. Al momento del lancio del comando C)omp il sistema compilerà il sorgente contenuto nella work-area, altrimenti chiederà il nome del file da compilare. Il risultato della compilazione è un file og-

getto registrato su disco con nome \*SYSTEM.WRK.CODE; si tratta del compilato in codice P-Code rilocabile che viene poi eseguito dal comando R)un.

Il compilatore può compilare anche codice BASIC. Si tratta di un BASIC "standard" abbastanza povero di istruzioni e funzioni (ma siamo nel 1979!). Il codice compilato è sempre P-Code e può essere linkato con il comando L)ink assieme ad altro codice P-Code per ottenere un unico eseguibile.

Il comando D)ebbug attiva un debugger interattivo dove è possibile settare breakpoint, esaminare il valore delle variabili, listare gli statement, etc... Certo è richiesto un certo tempo per imparare l'uso del debugger, ma questa è una cosa comune a tutti gli ambienti di programmazione.

Esaminando la lista delle funzioni di libreria disponibili e la strutturazione possibile del codice, oltre alla ricchezza di definizione per i tipi di variabile, ci si rende conto che nel 1980 questo linguaggio era il sogno di molti programmatori alle prime armi o provenienti da un BASIC fin troppo "basico" o da linguaggi come il FORTRAN con la sua rigida sintassi algoritmica.

```
File: L(dir E(dir R(em T(rans C(hng D(ate P(prefix K(rnch Z(ero U(cols Q(uit ?
APPLE1:
SYSTEM.PASCAL      45   3-Sep-85
SYSTEM.EDITOR      50   3-Sep-85
SYSTEM.FILER       30   3-Sep-85
SYSTEM.LIBRARY     38   3-Sep-85
SYSTEM.MISCINFO    1   25-Dec-83
SYSTEM.CHARSET     2   14-Jun-79
SYSTEM.APPLE       32   3-Sep-85
SYSTEM.SYNTAX      12   1-Jul-85
8/8 files <listed/in dir>, 216 blocks used, 64 unused, 64 in largest
```

### Come funziona il P-Code.

Una P-machine è una macchina (solitamente virtuale) in grado di eseguire le istruzioni del codice P-Code. Si tratta di una macchina a stack, cioè un sistema dove il paradigma dell'organizzazione a stack viene spinto al massimo livello.

In teoria non serve una CPU con registri, anche se all'atto pratico si può trarre vantaggio dalla presenza di uno o più registri generali. Quello che serve è un Program Counter (PC) e tre puntatori ad altrettanti stack, cioè aree di memoria specializzate. Una tipica P-Machine avrà i seguenti registri:

PC - Program Counter

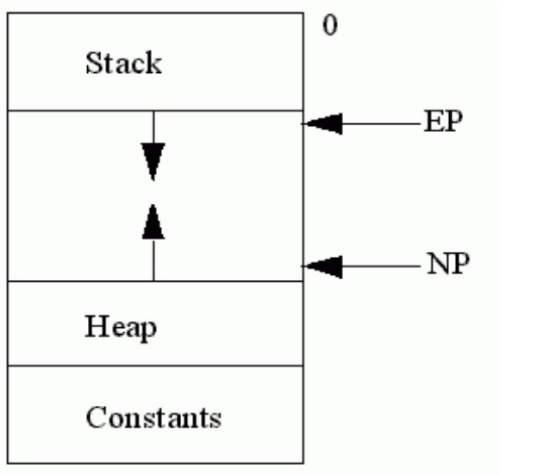
SP - Stack Pointer

MP - Mark Stack Pointer

NP - New Pointer

EP - Extreme Stack Pointer

L'area di memoria denominata Data Store è così organizzata:



L'area che contiene le costanti definite nel programma è generata dal compilatore ed è accessibile dal codice. Heap e Stack si disputano la memoria libera. Lo Heap è l'area uti-

lizzata dalle variabili mentre nello stack vengono inseriti i cosiddetti "Stack-Frame" che corrispondono ai livelli di annidamento delle chiamate presenti nel codice. Uno Stack-Frame è in effetti una struttura dati abbastanza complessa che contiene puntatori e parametri di chiamata alle funzioni.

Una istruzione in codice P-Code ha la seguente struttura:

OPC [T][P][Q]

dove OPC è il codice dell'istruzione (fino a 7 caratteri), seguito da un numero massimo di tre parametri opzionali:

[T] - indica il tipo di dati coinvolti nell'operazione; è una lettera, ad esempio A se è un indirizzo, B = dati booleani, I = dati interi, C = caratteri, R = reali, etc...

[P] - Indica il livello di stack delle variabili statiche (cioè globali).

[Q] - E' un intero o una etichetta.

Nel box nella prossima pagina un esempio di codice Pascal e del corrispondente compilato P-Code. L'esempio è tratto dal manuale utente del sistema WD-90.

### Conclusione.

Un sistema questo WD/90 che si pone fuori dalla normale concezione di un sistema di calcolo personale basato su microprocessore. Un esempio della interazione hardware/software che perseguendo una convivenza/ottimizzazione dei due campi perseguiva la massivizzazione delle prestazioni senza penalizzare troppo lo sviluppo con un codice troppo vicino all'architettura delle macchine, come appunto è l'Assembly.

(\*)



## Bibliografia e Riferimenti.

Western-Digital: WD/90 Pascal Microengine Reference Manual  
Western-Digital: WD/9000 chip set Technical Specifications  
[http://en.wikipedia.org/wiki/Pascal\\_MicroEngine](http://en.wikipedia.org/wiki/Pascal_MicroEngine)  
<http://www.cpushack.com/2014/02/09/western-digital-wd9000-pascal-microengine/>  
[http://en.wikipedia.org/wiki/P-code\\_machine](http://en.wikipedia.org/wiki/P-code_machine)  
<http://www.computerhistory.org/collections/catalog/102682825>  
<http://www.cpushack.com/2014/02/09/western-digital-wd9000-pascal-microengine/>  
[http://www.mwigan.com/mrw/WD\\_MicroEngine.html](http://www.mwigan.com/mrw/WD_MicroEngine.html)  
<http://www.pascal-central.com>  
<http://homepages.cwi.nl/~steven/pascal/book/pascalimplementation.html>

[Tutte le immagini usate nell'articolo provengono dai riferimenti citati]

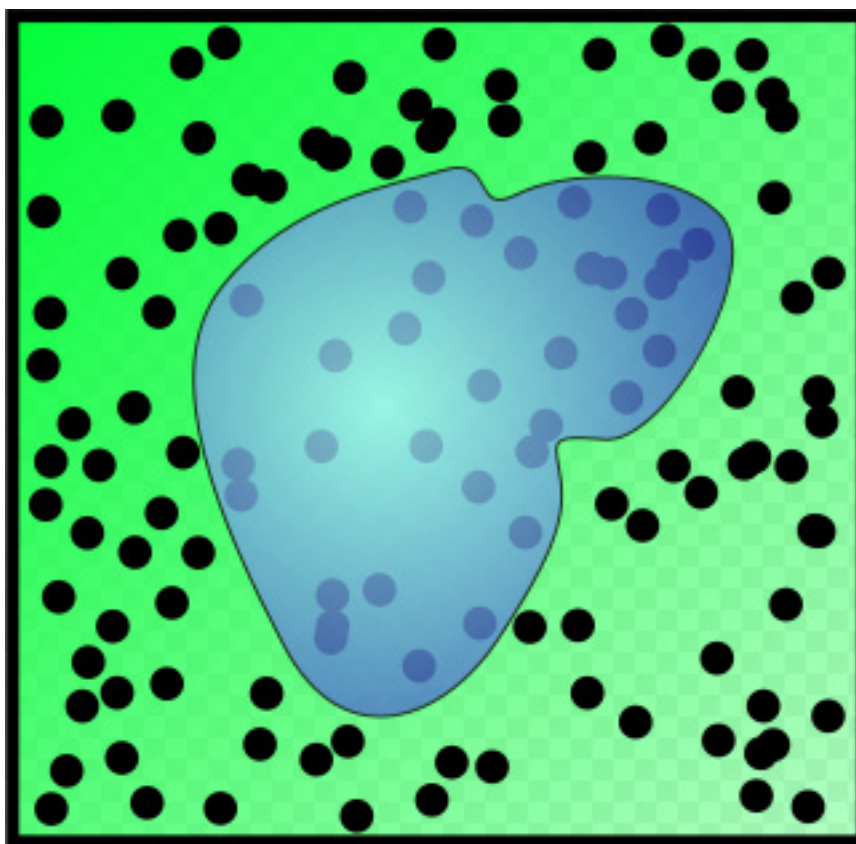
```
program locals(input, output);
  procedure addlocal;
    var i: integer;
        (*The variable i is local to the procedure addlocal*)
    begin
      i:=i+1
    end; (*of addlocal*)
begin
  addlocal
end.
```

===== codice P-Code risultato della compilazione =====

```
L 3          // Start of addlocal
  ENT 1 L 4
  ENT 2 L 5
  LODI 0 5    //The variable i retrieved from the first
              location above the mark stack i.e. 5
  LDCI 1
  ADI        // i+1
  STRI 0 5    // New value of i is stored back in 5
  RETP      // End of addlocal
L 4= 6
L 5= 7
L 6
I 10
ENT 1 L 7
ENT 2 L 8
MST 0
CUP 0 L 3    // Procedure addlocal is called
RETP
L 7= 9
L 8= 5
Q

I0
MST 0
CUP 0 L 6    // Main program block is entered
STP
Q
```

## Il metodo Montecarlo



*di Salvatore Macomer*

**I**l Metodo Montecarlo è un algoritmo che permette il calcolo di aree in maniera approssimata. L'idea è abbastanza semplice: se prendiamo un'area, cioè una superficie chiusa comunque delimitata da un contorno il più strano possibile e la inseriamo in una figura geometrica regolare, ad esempio un quadrato di lato sufficiente a contenerla tutta, allora la superficie dell'area da calcolare sarà una frazione dell'area del quadrato.

Finora abbiamo detto una banalità, cioè abbiamo affermato che l'area della superficie sconosciuta è sicuramente minore dell'area della figura regolare.

Ora supponiamo che il nostro sistema (quadrato + area all'interno di esso) sia all'aperto e cominci a nevicare, magari molto poco intensamente. Alcuni fiocchi cadono all'interno dell'area irregolare, altri fuori da essa ma sempre nel quadrato. Quelli che cadono fuori dal quadrato non ci interessano.

I fiocchi di neve che cadono sono un buon esempio di processo stocastico, cioè casuale, che è proprio quello che ci serve nel metodo Montecarlo.

L'area di un quadrato è facile da misurare ovviamente, la chiameremo  $A_q$ , mentre l'area della figura irregolare si chiamerà  $A_x$ . Consideriamo la seguente frazione:  $A_x/A_q$ . Dimosteremo con un semplicissimo ragionamento che questo rapporto è la probabilità che un fiocco di neve cada all'interno dell'area  $A_x$ .

Ci mettiamo a contare i fiocchi che entrano nell'area ( $N_x$ ) rispetto a tutti i fiocchi che cadono nell'intero quadrato ( $N_q$ ). Il fiocco di neve ha sicuramente probabilità 1 di cadere nel quadrato ma una probabilità  $< 1$  di cadere nell'area  $A_x$ . Notiamo ancora che se l'area  $A_x$  copre l'intero quadrato, allora  $N_x = N_q$  mentre se  $A_x = 0$ , cioè l'area incognita non esiste, allora  $N_x = 0$ ; inoltre se la super-

ficie incognita fosse metà di  $A_q$ , allora metà dei fiocchi cadrebbero al suo interno e l'altra metà fuori di essa.

La forma del perimetro dell'area irregolare non ha alcuna importanza. Anzi l'area da calcolare potrebbe essere anche formata da un certo numero di zone non connesse.

Insomma  $N_x/N_q$  tende per numeri grandi al rapporto  $A_x/A_q$ .

Ok, non è proprio una dimostrazione scientifica che se volete trovate sviluppata su Wikipedia, ma spero che la spiegazione sia convincente.

Siamo arrivati all'equazione:

$$A_x/A_q = N_x/N_q$$

e poiché  $A_q$  è conosciuta e sappiamo quanto valgono i numeri della nostra rilevazione della nevicata, ne segue che si può ricavare l'area  $A_x$  con la semplice equazione:

$$A_x = A_q * (N_x/N_q)$$

Cioè l'area incognita è uguale all'area conosciuta moltiplicata per la probabilità relativa (cioè un numero fra 0 e 1) di un processo stocastico che sia proporzionale al rapporto dimensionale delle due aree.

Non occorre che nevichi ovviamente, si può ottenere lo stesso risultato anche lanciando una manciata di sassolini nella zona del quadrato e poi componendo lo stesso rapporto. L'importante è che entri in gioco la casualità nella distribuzione degli elementi nell'area.

### Il "Metodo Montecarlo" è tutto qui!

Come si intuisce il suo nome deriva dall'introduzione di un processo stocastico tipico di giochi d'azzardo e il Principato di Monaco è noto per il suo Casinò!

L'uso del metodo è comune in tutte le situazioni matematiche o fisiche dove sia coinvolta un'area da calcolare e non sia possibile, o molto difficile, farlo con metodi analitici. Il classico esempio è quello dell'integrale definito di una funzione complicata. Come si sa l'integrale è l'area che sta "sotto la curva" ed è limitata dall'asse delle ascisse.

Trovare la primitiva potrebbe non essere possibile analiticamente. In questi casi si utilizzano metodi approssimati, come il metodo dei rettangoli che consiste nel dividere l'area in tanti rettangoli che in qualche modo cerchino di approssimare il più possibile precisamente l'area sottostante la funzione. Il metodo Montecarlo è parimenti preciso e molto spesso molto più alla portata di un calcolatore. Bisogna ricordare che essendo un metodo probabilistico la correttezza del risultato si ha solo se "ci sono molti lanci di dadi", cioè se i numeri  $N_q$  e  $N_x$  sono grandi. E' la famosa "legge dei grandi numeri".

Sapendo che l'area del quadrato si misura in  $A_q = L^2$  ( $L$  = dimensione del lato) e quella del cerchio come  $A_x = \pi * r^2$ , inscriviamo il cerchio nel quadrato (cioè il cerchio tocca i quattro lati del quadrato) e pertanto il raggio è pari a  $L/2$ .

Scriviamo l'uguaglianza vista prima:

$$A_x = A_q * (N_x/N_q)$$

$$\pi * (L^2)/4 = L^2 * (N_x/N_q)$$

e quindi ricaviamo pigreco:

$$\pi = L^2 * (N_x/N_q) * 4 / L^2$$

la dimensione del lato sparisce (si può semplificare fra i due membri) e si ottiene:

$$\pi = 4 * (N_x/N_q)$$

A questo punto basterà lanciare molte volte un sassolino e contare il rapporto  $N_x/N_q$  per avere il valore di pigreco.

Possiamo fare anche una ulteriore semplificazione: invece di considerare l'intero cerchio, consideriamone solo un quarto:

$$\begin{aligned} \pi \cdot (L^2) / 16 &= (L/2)^2 \cdot (N_x / N_q) \\ \pi &= L^2 \cdot (N_x / N_q) / 4 \cdot 16 / L^2 \\ \pi &= 4 \cdot (N_x / N_q) \end{aligned}$$

Incredibile, vero? La formula è la stessa e non poteva essere che così visto che pigreco uno è!

Nella realizzazione pratica con un calcolatore diventa essenziale il generatore di numeri casuali. Estrahendo due numeri fra zero e uno li associamo alle coordinate cartesiane X e Y individuando il punto "dove cade il sassolino".

Con la formula del cerchio che voi tutti ricorderete :-)

$$X^2 + Y^2 = R^2$$

si scopre se "il sassolino" è caduto dentro o fuori il cerchio (ma sempre nel quadrato ovviamente) quando la somma è uguale o inferiore a 1 (che è il quadrato del raggio).

Non occorre creare un programma vero e proprio, si può fare agevolmente con un foglio di calcolo come quello mostrato nella figura della pagina a fronte (abbiamo usato l'applicativo gratuito di Google Documents).

Le colonne A e B contengono rispettivamente l'estrazione per la coordinata X e per la coordinata Y. La formula nelle celle della colonna C ci dice se la coordinata del punto sta nel quarto di cerchio oppure cade nell'area esterna.

Una semplice formula da il risultato che viene mostrato nella figura nella cella A1 su sfondo rosso.

Se lo fate scoprirete subito che con pochi lanci si ottengono risultati molto vari e per nulla somiglianti al pigreco. Infatti per avere una stima valida di pigreco con questo metodo è necessario attivare molte simula-

zioni. Con 10.000 lanci abbiamo ottenuto un 3,14 ma il terzo decimale è ancora sbagliato. Ovviamente fare molte iterazioni non è un problema per un calcolatore!

### Conclusioni

Il calcolo, o per meglio dire la stima, del valore della costante pi-greco ci è servita come esempio per illustrare le idee del metodo Montecarlo e come si possa implementare (o meglio simulare) con un calcolatore. Ci siamo valse di un foglio di calcolo ma ovviamente con la programmazione si può ottenere una stima molto più attendibile aumentando molto il numero di iterazioni.

$$f_x = 4 * (D1 + I1 + N1 + S1 + X1) / (E1 + J1 + O1 + T1 + Y1)$$

	A	B	C
1	3.146746747		
2	X	Y	X^2 + Y^2
3	0.7159773454	0.7817817028	1.12380619
4	0.1813606292	0.02866711117	0.0337134811
5	0.6838757519	0.2064754899	0.510318172
6	0.6038326469	0.9816287752	1.328208918
7	0.1040989405	0.491742033	0.2526468164
8	0.4763531408	0.6728848428	0.6796863264
9	0.7154223558	0.4756485987	0.7380707367
10	0.1224249188	0.9402102605	0.8989831947
11	0.6487066609	0.293178694	0.5067740785
12	0.6926232132	0.4655528074	0.696466332
13	0.4494997902	0.5373225729	0.4907656088
14	0.6468443164	0.110586303	0.4306369001

# Automatik (23)

## Il lato oscuro



*Di Lorenzo Paolini*

*Dove si descrive come certe aziende vivano più per l'apporto "in nero" dei dipendenti che per la dedizione e l'ingegno del titolare.*

*Quando fui assunto dall'Automatik, una azienda artigiana che si occupava di distribuzione di videogiochi negli anni '80, ero ben conscio che non andavo a lavorare per una società internazionale. La dimensione piccola, familiare dell'artigianato in genere ha (o almeno aveva quando ci sono entrato io in quel mondo) pochi vantaggi e molti, moltissimi svantaggi.*

*Un vantaggio era il rapporto umano che si veniva a creare fra titolare e dipendenti. Nel caso della Automatik di proprietà di Romano, la dimensione familiare era incancrenita addirittura. La moglie faceva lavoro da ufficio mentre i due figli, adolescenti all'epoca, erano dei rompiscatole sempre in giro a curiosare e quindi indirettamente a controllare se*

*lavoravi o meno.*

*Ovvio che i due ragazzi, la femmina sui 16 anni e il maschio attorno ai 12, erano ovviamente attratti dai videogiochi e dai flipper che non mancavano mai nel laboratorio annesso alla casa di abitazione che era anche sede della ditta. Il maschietto era una autentica spina nel fianco per noi dipendenti, perché combinava qualche guaio e perciò bisognava continuamente tenerlo d'occhio. La ragazza era più simpatica anche se un po' timida mi sembrava (poi avrei dovuto ricredermi) ma certo era sempre "la figlia del capo tribù".*

*Gli svantaggi invece erano molti. A parte il fiato sul collo continuo dovuto alla presenza di qualcuno della casa sempre in giro, l'aspetto negativo principale era proprio quello strutturale. Male equipaggiati e organizzati si doveva in qualche modo "sposare" l'azienda se si voleva sopravvivere decentemente.*

*Quando parlo di decenza intendo la dimen-*

sione di respiro che nasce sul lavoro qualora si riesca a ritagliarsi dei propri spazi di libertà.

Provenendo da un paio di esperienze di lavoro nel commercio, trovavo sorprendente che l'azienda dove ero capitato potesse prosperare a lungo. Intendiamoci: capii subito che soldi ne giravano, ma era la strategia a medio-lungo termine che mancava del tutto!

L'idea che si potesse aumentare gli introiti semplicemente senza adottare una seria strategia di pianificazione ed investimenti adeguati, derivava al titolare signor Romano, dalla sua esperienza più che trentennale nel settore e che si traduceva in azioni semplici, quasi elementari: mettevi giù un gioco in un locale pubblico, passavi ogni quindici giorni a ritirare l'incasso e ogni due mesi a cambiarlo facendo ruotare l'asset aziendale fino all'esaurimento fisico (letteralmente!) delle macchine.

Gli acquisti si limitavano ai pezzi di ricambio necessari e all'acquisto di altrettanti giochi per rimpiazzare quelli che esaurivano il loro compito. Per una strana visione del mondo e per l'attaccamento che aveva Romano alle cose sue, i vecchi "catorci" mica li portavi in discarica! Tutt'altro! Così la ditta affittava piccoli magazzini in giro per la periferia dove le macchine esaurite (diciamo pure scassate) riposavano i loro ultimi anni. Solo dopo la mia presenza si riuscì in qualche modo ad utilizzare i vecchi cabinet per nuovi giochi entrando in quel mercato "delle piastre" dove si comprava l'elettronica e il resto in qualche modo si adattava.

Ma il mondo non andava alla stessa velocità di Romano, anzi! All'inizio dell'80 ci fu il boom dei videogiochi giapponesi e americani e il tran-tran della azienda Automatik ne venne sconvolto. Si rendevano necessari nuovi capitali e nuove strategie. Romano non era uno stupido e l'aveva capito che era meglio se si metteva a correre anche lui. Solo che

pensava si potesse fare a basso costo. Le classiche "Nozze con i fichi secchi" insomma!

Il sottoscritto fu assunto perché per studi e passioni ero un tecnico elettronico con qualche infarinatura digitale-informatica (non dimentichiamo che nel 1980 praticamente nessuno aveva un calcolatore personale e quindi crearsi un background nel settore era tutt'altro che facile!). Il problema fu che vuoi per una mia timidezza e vuoi per l'isolamento culturale in cui sguazzava Romano, non si partì con il piede giusto. Romano pensava di prendere una persona qualificata e fargli fare tutto: facchinaggio, logistica, incassi,... ma anche (ed ecco l'alzata di ingegno) riparazioni e assemblaggio di nuovi giochi. Non capiva il tapino che dedicarsi al laboratorio elettronico escludeva il resto, non fosse altro che per mancanza di tempo. Da parte mia ero pavido nel chiedere che si cambiasse registro, anche perché il mio progetto era schiodarmi dal lavoro precedente che nulla aveva a che fare con l'elettronica che sentivo era la mia strada, e dimostrare che non facevo una stupidaggine come sosteneva mio padre il cui consiglio era stato di lasciar perdere e non rinunciare al posto fisso.

Se fossi tornato con la coda fra le gambe il mio amato genitore dalla mentalità retrograda ne avrebbe goduto e i "te l'avevo detto" si sarebbero sprecati!

Quindi mi adattai a fare il factotum. Bisogna che riconosca per onestà intellettuale che non ero così certo di essere in grado di sviluppare la parte laboratorio elettronico dell'azienda: in fondo avevo una esperienza hobbistica e una base teorica scolastica. Per tali ragioni giudicai prudente non impuntarmi a ritenermi tecnico specializzato e aspettare l'evolversi della situazione.

Di fatto quindi il mio arrivo in azienda non coincise con quel "salto di qualità" che sarebbe stato necessario per tenerla a bada

*l'agguerrita concorrenza. I profitti incrementarono ma per un effetto contingente: erano anni d'oro in fondo, bastava avere macchine da posizionare nei locali più impensati e il gettito mensile, anche se non paragonabile ai milioni di Lire che "tiravano" i posti migliori, comunque arrivava e copriva abbondantemente le spese.*

*Il tranquillo menage aziend-famigliare proseguì inperterrito con la signora, moglie del titolare, che teneva l'ufficio e il marito che perdeva tempo... In fondo aveva lavorato "una vita" come amava dire ed era anche vero, visto che aveva iniziato a 14 anni sotto le rigide grinfie di un parente che lo "aveva tirato su dritto come un fuso..."*

*I retroscena gestionali mica li potevo immaginare da solo! Fu Damiano che un poco per volta me ne rese edotto. In pratica in azienda c'erano due contabilità: quella fiscale ad uso e consumo di eventuali ispezioni della Guardia di Finanza (che fra l'altro in quasi tre anni di permanenza io non ho mai visto mettere piede in ditta) e il classicissimo "nero", cioè quella vera, ovviamente senza esposizione ufficiale. Risultava pertanto che se incassavi in un mese una media di dieci/quindici milioni di Lire (quello era più o meno il gettito), nei corrispettivi ci avresti trovato più o meno la metà! Tolte le spese, le tasse e quant'altro lasciavano un buon stipendio per il titolare, ma il guadagno era altrove... Una situazione che con il senno di poi definisco "onesticchia", dato che le finezze che vidi poi nel corso della mia vita lavorativa per non pagare tutte le tasse dovute, andavano ben oltre!*

*Il resto dell'incasso dove finiva? In mille rivoli che ormai sfuggivano addirittura agli stessi titolari (marito e moglie). A parte una cassaforte nascosta in cantina che non ebbi mai l'occasione di vedere nemmeno da chiusa, altri contanti erano in certe cassette di sicurezza presso banche, comprese quelle in Austria che non distava moltissimo, giusto il viaggetto di una giornata andata e ritorno.*

*Da certe frasi e confidenze che mi fece il ti-*

*tolare durante gli spostamenti che si facevano assieme, soprattutto nei primi tempi quando ancora io "dovevo imparare il mestiere", una parte degli incassi venivano nascosti anche alla moglie. Romano aveva cioè una sua propria contabilità (e sono tre) e peraltro non posso escludere che anche la sua consorte praticasse in questo campo e quindi il dare-avere forse aveva dimensione quattro...*

*Quello che mi raccomandò Daniele e che io osservai scrupolosamente, era di non prendere mai soldi "apparentemente" abbandonati sui tavoli o in macchina o ancora da computer nell'ufficio. Mi spiegò che Romano era "furbo" e tendeva delle trappole. Infatti qualche dipendente passato da lì ci era cascato e in quel caso Romano non aveva nessuna remora: licenziamento in tronco.*

*Mi ritrovai in seguito, anche per altre vicende personali, ad osservare come certe persone siano inflessibili e non arretrino di un passo sull'onestà dei loro dipendenti che vogliono totale e cristallina e poi siano invece i primi a "tramacciare" per fregare lo Stato e spesso gli stessi dipendenti...*

*Damiano mi spiegò la sua di contabilità (e cinque!). In pratica anche lui teneva una cassa che fisicamente era una scatola di ferro in uno dei magazzini della ditta, proprio il più scassato ed abbandonato. Ci metteva dei soldi e li usava per le "spese generali", come diceva lui. Cioè ci comprava attrezzatura, materiale di ricambio per i giochi che Romano si rifiutava di sostituire perché "... era ancora buono...", rimborsi spese quando andava a mangiare e non era in servizio ufficiale, etc...*

*Il meccanismo di alimentazione della "Cassa del Mezzogiorno" come la chiamammo ad un certo punto, era semplice e sorprendentemente efficace: prendendo le monete che "saltavano fuori" dalle gettoniere nel loro percorso verso la scatola a loro destinata, si potevano fare due/tre mila Lire per ogni gioco. Non erano soldi di Romano in fondo perché non erano il corrispettivo per la partita; diciamo che erano le perdite dei giocatori che si vedevano*



qualche volta “mangiate le duecento lire”. A volte li rifondeva il proprietario del locale e se questi lo faceva presente gli si restituiva ciò che aveva anticipato. Spesso però il titolare del bar non diceva nulla o perché non aveva la percezione di quanto ammontasse il totale che aveva restituito agli avventori o perché lo considerava una perdita di esercizio, peraltro ben ripagata dagli incassi (che nemmeno lui registrava fra i corrispettivi, ovvio!).

Quando durante una giornata ritiri una decina di incassi il risultato è che ti ritrovi con una ventina di mila Lire che “non sono di nessuno”, erano cioè della “Cassa per il Mezzogiorno” e visto che a volte con quel fondo andavamo anche al ristorante, il nome era più che appropriato!

All'epoca con duemila/tremila Lire mangiavi una pizza, tanto per capire di cosa stiamo parlando e con ventimila portavi la ragazza a cena in un buon ristorante!

Daniele si era preso a cuore quel “ragazzo venuto dalla campagna”, cioè io. Un po' la mia innata simpatia (lo dicono gli amici) e un po' la sua solitudine (manco era fidanzato), lo indussero a creare con me un'amicizia che è l'unica cosa buona che mi è rimasta dalla mia esperienza in ditta Automatik (magari ne salvo anche un'altra, devo avervelo già raccontato :-).

Inutile dire che il sodalizio funzionò benissimo anche con la cassa comune e la contingenza, cioè il vero e proprio boom dei videogiochi da bar, fecero il resto. Avevamo in cassa qualche milioncino e se la cosa poteva imbarazzare sul fronte morale, dall'altro punto di vista e cioè: la ditta deve pur funzionare se vogliamo conservarci gli stipendi, era una vera assicurazione. Con quel fondo comprai addirittura un oscilloscopio che tenevo a casa non giudicando opportuno che stesse in un deposito polveroso. Un giorno l'avevo portato al laboratorio per fare una certa riparazione ad un paio di monitor video e Romano rientrando inaspettatamente (lo faceva spesso) mi sorprese con “l'aggeggio” e volle sapere. Avevo

la scusa pronta, ovviamente e gli dissi che era mio e che l'avevo portato da casa perché senza oscilloscopio riparare certi guasti era impossibile.

Pensavo facesse spallucce come sempre ma quella volta Romano si rivelò “un signore”, cosa rarissima peraltro, e infatti chi non ricorda l'eccezionale ondata di freddo attorno al 1980? Anche la Natura volle sottolineare con un avvenimento epocale quella fatidica data!

Romano addirittura pretese che ne comprassi uno a spese della ditta e non battè ciglio nemmeno quando gli dissi che “uno di quegli aggeggi” costavano più di un milione.

Insomma ebbi due oscilloscopi e fu lo stesso Daniele a dirmi che quello “vecchio” (si fa per dire) potevo portarmelo a casa e ce l'ho ancora.

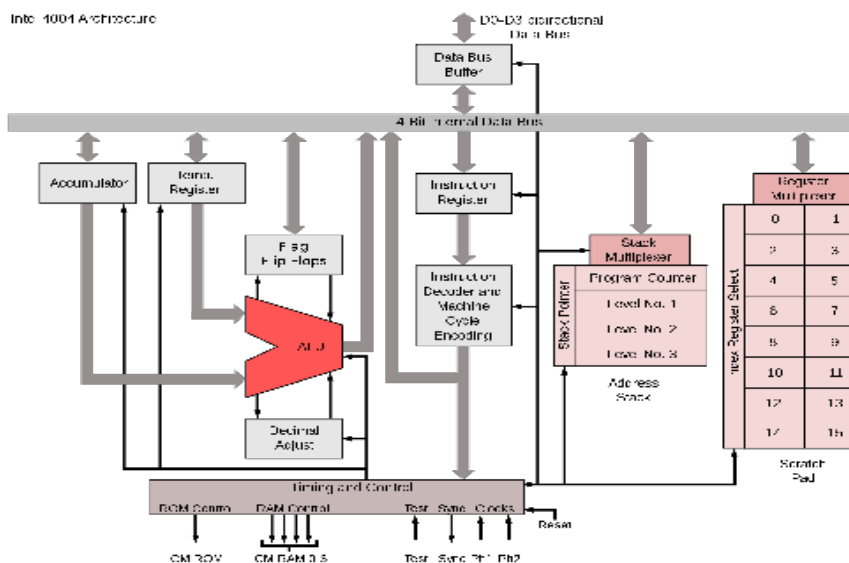
La Cassa del Mezzogiorno era un motore abilitante per l'economia (si direbbe adesso). La possibilità di investire letteralmente in materiale di qualità, ad esempio sostituendo i joystick quando cominciavano a dare segni di cedimento senza aspettare che si rompessero del tutto, innescava un circolo virtuoso: i giochi andavano meglio, i clienti spendevano di più, avevi meno chiamate per riparazioni e potevi impiegare il tempo a rinnovare altri giochi, e a cambiarli più spesso con maggiore resa.

L'aumento degli incassi venne registrato anche in ditta e dopo un anno dalla mia assunzione Romano ce lo disse una sera che era particolarmente di buon umore. La ditta andava bene e il merito era anche nostro. In realtà il merito era TUTTO nostro ma non ritenni opportuno puntualizzarlo :-)

Comunque guadagnammo un premio: ci mandò in vacanza a Rimini per un weekend lungo, tre giorni tutto pagato. Partimmo il venerdì pomeriggio e tornammo il lunedì successivo. Ovviamente nulla era gratis nella ditta Automatik e il lunedì dovevamo passare per Faenza a prendere un paio di giochi nuovi.

... ma questa è un'altra storia...

# Programmare l'Intel 4004



di Tullio Nicolussi

Il microprocessore Intel 4004, del quale raccontiamo la storia della sua creazione in questo fascicolo, è considerato il primo microprocessore commerciale che divenne disponibile a cominciare dal 1971.

La Intel vendeva il processore come componente di quello che veniva chiamato Micro Computer System con sigla MCS-4, formato da quattro chip: 4001 la ROM, 4002 la RAM, 4003 shift register e 4004 la CPU.

Dallo schema funzionale che si vede in apertura, si capisce che la sua struttura è un po' diversa da quella classica degli 8 bit che verranno subito dopo. C'è l'Unità Aritmetico Logica (ALU) e l'accumulatore. Il Program Counter occupa una locazione dell'area di stack che ne dispone di quattro in totale. Questo significa che sono possibili solo tre livelli di salto a subroutines, se ne arriva un quarto il più vecchio inserito si perde.

Questa è una prima importante differenza rispetto ad una CPU di "seconda generazione".

Le operazioni aritmetiche e logiche vengono svolte coinvolgendo due elementi: Accumulatore e registro temporaneo. Significa che non ci sono registri fissi da nominare con le lettere B, C, etc... come si trovano

nell'8080 o nello Z80 (tanto per citarne due fra i più noti), ma si dovrà operare partendo da un multiplexer di registri (16) che formano quello che viene chiamato Scratch Pad, una area di lavoro insomma!

Il clock standard è pari a 750 KHz e ogni istruzione ne richiede un numero variabile da 8 a 16. Una istruzione semplice viene eseguita in 10,8 microsecondi mentre una istruzione estesa richiede 21,6 microsecondi.

Il sistema MCS-4 prevede che le istruzioni siano nella ROM e che vengano svolte secondo una sequenza fissa che è la seguente (per una istruzione a 8 cicli):

1. emissione dell'indirizzo attraverso la sequenza di tre serie di valori sul bus dati che è da 4 bit (necessari tre cicli di clock);
2. con l'indirizzo a 12 bit si prelevano 4 + 4 byte dell'istruzione (due cicli di clock necessari);
3. esecuzione dell'istruzione negli ultimi tre cicli di clock.

Il totale di istruzioni macchina è di 45 e si possono dividere in tre gruppi: 16 istruzioni macchina vere e proprie, 14 operazioni sull'accumulatore e sullo stack e 15 operazioni che interagiscono con I/O e memoria.

In realtà, grazie al codice OPA modifica-

tore, tutte o quasi le 256 combinazioni di bit sono sfruttate. Molte istruzioni possono agire su qualsiasi registro. Ad esempio per incrementare il contenuto di un registro si userà

*INC, R* (Con  $R = 0...15$  che sta ad indicare appunto il registro coinvolto).

Le istruzioni possono essere singole (un byte) o doppie (2 byte) e sono organizzate in gruppi di quattro bit (ricordiamo che il 4004 ha una lunghezza di parola di 4 bit, appunto). I primi 4 bit di una istruzione (chiamati *OPR*) sono il codice dell'istruzione stessa (ad esempio *ADD*, *SUB*, etc...), mentre i successivi 4 bit (chiamati *OPA*) possono contenere un valore immediato o un riferimento ad un registro o un modificatore dell'istruzione. Nel caso di istruzioni a 16 bit l'*OPA* di ciascun byte, può contenere un indirizzo di memoria che coinvolge l'operazione stessa.

L'area denominata "Scratch Pad" costituisce una risorsa di 16 registri a 4 bit utilizzabili anche in coppie (8 registri doppi). L'indicazione di utilizzo di uno di questi registri avviene all'interno della porzione di codice *OPA* dell'istruzione indicando semplicemente il numero (da 0 a 15 o da 0 a 7 se in coppia) del registro coinvolto. Di fatto quindi questo micro di prima generazione traccia la strada per l'ingegnerizzazione degli sviluppi successivi e l'utilizzo dei registri temporanei ne è la prova più evidente.

L'elenco delle istruzioni è limitato ma va anche considerato che si tratta di un sistema progettato esplicitamente per diventare il motore di calcolatrice e quindi essenzialmente per fare operazioni matematiche.

Troviamo la classica istruzione per il caricamento immediato nell'accumulatore:

*LDM, DDDD*

con il codice *LDM* inserito nei primi 4 bit (*OPR*) dell'istruzione letta dalla ROM e i 4 bit di dati (*DDDD*) nella parte *OPA* dell'istruzione.

Il caricamento dell'accumulatore partendo da un registro:

*LD, RRRR*

Dove *RRRR* è il numero (da 0 a 16) della locazione dello Scratch Pad che si vuole usare.

Si possono scambiare i valori dell'accumulatore con un registro attraverso l'istruzione

*XCH, RRRR*

Aggiungere il valore di un registro all'accumulatore:

*ADD, RRRR*

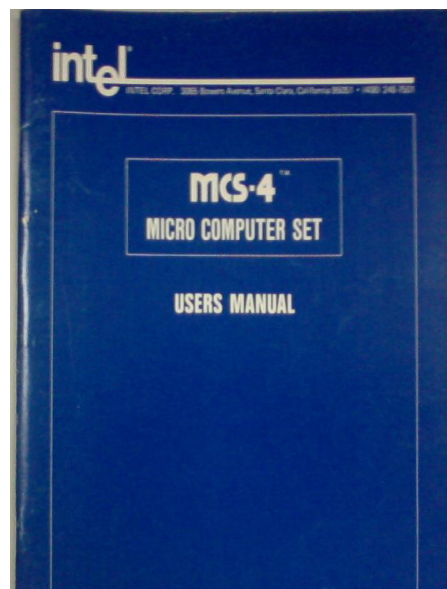
o sottrarlo:

*SUB, RRRR*

Le istruzioni che agiscono sull'accumulatore appaiono abbastanza complete: si può complementarne il contenuto, azzerrarlo, incrementarlo e decrementarlo, ruotare i bit a destra e a sinistra,...

Le istruzioni tengono conto del valore del Carry che può essere manipolato (settato o resettato) da apposita istruzione.

Più macchinoso è l'interazione con la RAM,



infatti è necessario scegliere il banco coinvolto, poi caricare l'indirizzo in un registro specifico del chip 4002, abilitare il segnale di lettura o scrittura e infine utilizzare le relative istruzioni di store/read.

Ad esempio per aggiungere due numeri in notazione scientifica (16 digit di mantissa e due di esponente) registrati nella RAM, servono la bellezza di 42 istruzioni.

Nei box colorati in queste pagine due esempi di scrittura di codice. Nel primo una subroutine lavora su due registri, il registro 0 e il registro 1. Essa effettua un loop sul contenuto del registro 1 fino a che non viene settato il carry; a questo punto il contenuto del registro 0 viene incrementato e la routine finisce.

Nel secondo esempio, più complesso, si interagisce con la RAM. Notate come sia necessario indicare esplicitamente che si eseguirà una scrittura (istruzione WRR), successivamente con WPM la scrittura viene eseguita e lo switch di scrittura viene riportato a false da una nuova esecuzione di WRR.

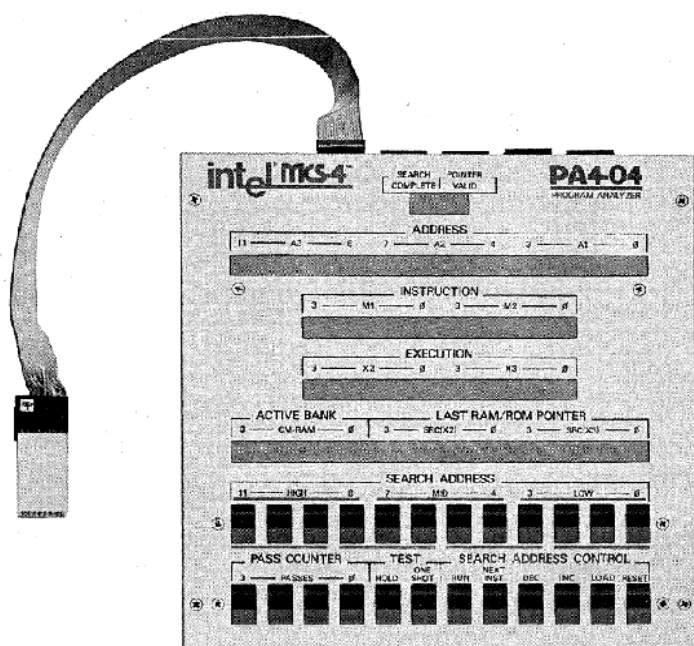
Sappiamo che i successivi progetti di microprocessori hanno inglobato i meccanismi di interazione con le memorie esterne direttamente nella logica programmata.

Lo sviluppo dell'Intel 4004 ha evidenziato il fatto che non si era pronti ad utilizzare un kit di questa potenzialità ma anche complessità di implementazione nei progetti di automazione. La Intel si è preoccupata di

questo, producendo manuali ben documentati ed esempi di utilizzo ma anche strumenti di valutazione e di sviluppo. E' il caso dell'apparecchiatura siglata PA4-04 (Program Analyzer for MCS-4).

Si tratta di una periferica che si aggancia sopra il micro 4004 con una clip e ne esamina i segnali su un display esterno.

Questa apparecchiatura, direi indispensabile per lo sviluppo di codice, permette di settare un breakpoint raggiunto il quale si potranno rilevare i valori dei registri della CPU e il suo stato e anche eseguire il codice passo-



LABEL	CODE	OPERAND	COMMENT
IN,	XCH	1	/ Scambia il Reg. 1 con Accumulatore
	IAC		/ Incrementa l'accumulatore con carry
	XCH	1	/ Scambia Accumulatore e registro 1
	JCN	10 NC	/ Salta se Carry = 0
	INC	0	/ Incrementa il registro 0
NC,	BBL	0	/ ritorno da subroutine

passo.

Successivamente Intel fornì anche schede preassemblate contenenti l'intero sistema MCS-4, pronte per essere inserite nel bus di apparecchiature di controllo. La serie Intellect-4 è invece un sistema chiavi in mano completo di unità centrale, memoria espansa a 5 Kbyte interfaccia per TTY e pannello frontale di controllo.

### Conclusione.

Il microprocessore Intel 4004 è il primo esempio di una logica programmabile "general purpose" e come tale ha gettato la basi per lo sviluppo successivo dei microprocessori a 8 e 16 bit che hanno dominato il decennio degli anni '80. L'introduzione del microprocessore a 8 bit (8008) da parte della stessa INTEL non bruciò affatto il mercato del suo "fratellino" minore che rimase a listino almeno fino al 1975.

(\*)

LABEL	CODE	OPERAND	COMMENT
STR,	FIM	0P 224	
	SRC	0P	/ Select ROM port 14.
	LDM	1	
	WRR		/Turn on write enable.
	JMS	COM	/ Routine COM sets up PRAM / address.
	LD	2	/ High 4 data bits to accumulator.
	WPM		/ Write to PRAM
	LD	3	/ Low 4 data bits to accumulator.
	WPM		
	FIM	0P 224	
	SRC	0P	/ Select ROM port 14.
	CLB		
	WRR		/ Turn off write enable.
	BBL	0	/ Return to program

### Bibliografia:

Faggin, F. ; Shima, M. ; Hoff, M.E. ; Feeney, H. ; Mazor, S.; *THE MCS-4 - An LSI micro computer system*; IEEE Solid-State Circuits Magazine, V. 1, Issue 1; DOI: 10.1109/MSSC.2009.931980.

INTEL: *MCS-4 Assembly Programming Manual*.

INTEL: *MCS-4 User's Manual*.

INTEL: *MCS-4 Data Sheet*.

INTEL: *MCS-4 Kit (brossure)*.

INTEL: *PA4-04, Program Analyzer*.

INTEL: *MCS-4 Hardware Simulator*.

INTEL: *Data Catalog 1973*.

## La zampata del vecchio leone



### Di Tullio Nicolussi

**I**l signore che appare nello screen di apertura, tratto da un programma della BBC, è Sir Clive Sinclair. Irriconoscibile a prima vista, vero?

In realtà confrontando la sua immagine con quelle risalenti ai suoi “anni ruggenti”, quando presentava alla stampa le sue meraviglie elettroniche che si chiamavano Spectrum e QL, ma anche la sfortunata televisione portatile e l'improbabile macchinetta elettrica C1, i tratti dell'uomo si riconoscono.

Che vita fa adesso il protagonista della rivoluzione digitale della Gran Bretagna e oserei dire per buona parte della intera Europa? Vive abbastanza modestamente a giudicare dall'arredamento della sua casa, visibile nel filmato di questa intervista recente. I suoi milioni se ne sono andati, forse per buona parte sprecati nei progetti fal-

limentari sopra accennati, ma comunque cerca di fare l'imprenditore e viene spesso interpellato dai giornali inglesi per quanto riguarda l'evoluzione tecnologica.

In questa ultima intervista Sinclair si lascia andare ad una dichiarazione per qualche verso sorprendente. Dichiarò infatti che l'evoluzione del personal computer è deludente. Infatti - sostiene - rispetto a quanto stato fatto nei primi anni '80 nel mondo e da lui stesso, i progressi successivi sono stati poveri di innovazione riducendosi ad una mera evoluzione tecnica senza creatività. Si sarebbe aspettato - continua - di avere già nel 2000 o poco dopo, delle macchine intelligenti che potessero sostituire il medico o l'insegnante. Non è stato così e secondo l'inventore dello ZX80, la colpa è nella mancanza di fantasia e ingegno da parte di coloro che dopo essersi impossessati



del mercato non hanno poi saputo portare al pubblico la vera rivoluzione che tutti si aspettavano.

E' un punto di vista che ha del vero, ovviamente. E' evidente a tutti che il mercato che è andato nella direzione del PC IBM compatibile ha affossato qualsiasi velleità di innovazione. Però è anche vero che quello è un mercato e null'altro. Cioè raggiunto un certo livello di usabilità nei campi dell'elaborazione "spicciola" (contabilità, scrittura, immagine,...), non c'era motivo di stravolgere il progetto. Invece si sono aperti nuovi filoni che hanno ricavato dall'invenzione del personal computer il loro seme generativo. Pensiamo alla telefonia mobile (che di telefonia oggi ne fa poca ma fa tutto il resto...), alle console per videogiochi, alle tv smart, ad Internet stessa e alle macchine che la fanno funzionare, al WI-FI, etc...

Non si può dire che la rivoluzione informatica dei primi anni '80 sia rimasta al palo!

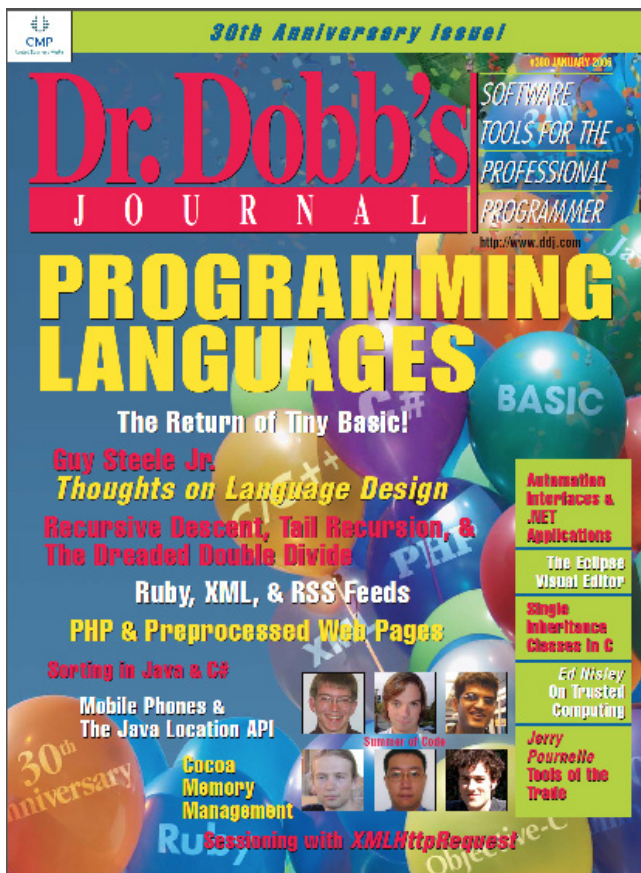
Oggi Clive Sinclair sta investendo, non è dato sapere quanto, nel progetto di creare una console da gioco che assomiglia esternamente in tutto e per tutto al vecchio Spectrum. Per ora si parla di retro-console, cioè qualcosa che emula lo Spectrum e ci fa girare i giochi. Immaginiamo ci sia dentro il

solito ARM e qualche componente di interfaccia, tipo RaspberryPI... Se ci sarà una evoluzione più cogente sarà tutto da vedere.

Fra l'altro speriamo che questo nuovo progetto dello Spectrum non abbia ereditato la famigerata tastiera a membrana...

Certo viene da pensare che dopo aver espresso la sua delusione per la mancanza di creatività, poi il "vecchio leone" accenna ad una zampata che assomiglia all'approccio indeciso di un gattino più che all'affermazione di potenza da vero leone...





## The rise of Dr. Dobb's

di Sonicher

Un'altra storica testata della storia informatica mondiale ha scelto di ridimensionare la sua presenza sul Web, preludio forse della chiusura addirittura della testata.

Lo comunica mestamente Andrew Binstock, uno degli editor della rivista in un post del Dicembre 2014.

*"After 38 years of glory, the long run of Dr. Dobb's has come to an end."*

Questo l'incipit del messaggio, vediamo di capirne i motivi. Non si tratta di scarso interesse da parte del proprio pubblico; infatti le visite al sito sono cresciute più o meno di un milione all'anno, arrivando a superare nel 2014 i dieci milioni di pagine visitate. Trattandosi di un sito molto verticale, cioè dedicato agli sviluppatori, e che non indugia in effetti speciali per attrarre nuove visite, la cosa appare sorprendente - affer-

ma Andrew - e conferma la necessità per la comunità di sviluppatori di trovare informazioni tecniche in merito allo sviluppo del software che siano aggiornate e attendibili.

I numeri sembrano non impressionare la catena editoriale che ha come vertice una società che si chiama United Business Media e che avrebbe deciso un "sunset" della web-testata. In realtà il significato del termine implica semplicemente che non ci saranno aggiornamenti di contenuto. Ma questo - spiega Andrew - significa in poche parole "chiudere", anche se non immediatamente.

Il mondo della pubblicità sul Web è cambiato negli ultimi anni e ora il ricavo pubblicitario del sito in questione è appena il 30% di quello che era 18 mesi fa. In pratica gli investitori hanno realizzato che la pubblicità sul Web è molto meno efficace e quindi deviano gli investimenti verso altri



canali più efficaci. Quali siano questi canali non sono in grado di dirlo non essendo del settore; forse i social...

Altre iniziative editoriali che hanno la loro vetrina su Web si salvano organizzando eventi e conferenze tematiche. Il problema è che DrDobb's si occupa di un settore troppo frammentato e non sarebbe profittevole mettere su piccole conferenze specifiche quando i grandi temi (Java, Google, etc...) sono già coperti dai maggiori protagonisti del mondo dello sviluppo.

Quindi l'editore ha deciso che prima di arrivare ad un bilancio negativo è meglio congelare il sito, pur conservandone il contenuto attuale. C'è da dire che gli articoli erano pagati e non frutto di semplice volontariato. Negli States vige questo principio che mi sembra sacrosanto: chi lavora va pagato.

Andrew ci descrive la sua amarezza di autore cresciuto a pane e DrDobb's, come dice lui, studiando tutti gli argomenti sulle pagine della rivista e imparando nuovi linguaggi e le tecniche programmatorie che via via i fascicoli riportavano.

L'avvento del Web ha amplificato la missione della testata ed ha creato una comunità che volentieri ha discusso e condiviso idee e codice.

Rimane però l'amarezza del momento - dice ancora Andrew - di non poter proseguire per la stessa strada dopo tante battaglie e tanti successi.

La conclusione dolce-amara sta in tre parole: "Grazie a tutti".

Riferimenti:

<http://www.drdoobbs.com/architecture-and-design/farewell-dr-dobbs/240169421>

## Conclusioni personali.

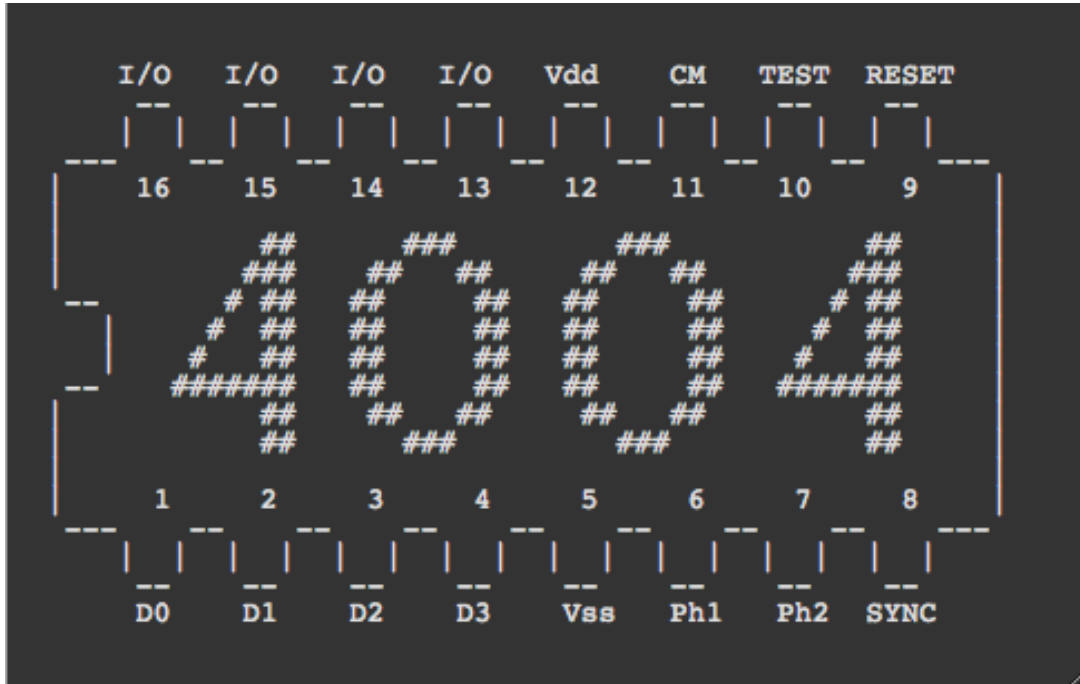
Quello che salta subito all'occhio è che se un sito da dieci milioni di pagine viste in un anno non riesce a sopravvivere sul Web, allora c'è pochissima speranza per gli altri... Il modello di business basato sulla raccolta pubblicitaria evidentemente ha delle lacune che non sono facilmente colmabili.

Quale potrebbe essere la strada in mancanza di uno sponsor "illuminato"? Qualcuna mi viene in mente ma sono tutte incerte e di difficile attuazione. Non sono poi un esperto di marketing e quindi assumo che se non ce l'hanno fatta gli editori americani forse non c'è una soluzione.

Certo che se nemmeno gli americani sono disposti a scucire un dollaro per accedere ad informazioni professionali, vuol dire che siamo tornati indietro e che la ricchezza di notizie sul Web equivale a non averne nessuna.

(-)

## Sistema che vai...



di Lorenzo/2

... emulatore che trovi!

Infatti quasi tutte le macchine di calcolo del passato hanno il loro bravo emulatore. Non fa eccezione il primo microprocessore della Intel, quel 4004 al quale contribuì la gloria nazionale Federico Faggin.

Se per un sistema di calcolo l'emulazione potrebbe essere un esercizio di tecnica programmatoria, per i microprocessori l'emulazione è nata assieme ad essi e anche oggi, forse di più del passato, l'emulatore è un ambiente che consente ai tecnici ed ingegneri prendere confidenza con le prerogative di un sistema magari di nuova introduzione. Nel comparto educational a maggior ragione è importante emulare sia per i costi che un vero sistema di sviluppo "fisico" comporterebbe se si volesse mettere a disposizione di ogni allievo, e sia per la flessibilità che esso

consente.

La potenza del linguaggio Javascript consente oggi di sviluppare emulatori che girano direttamente nel browser, senza bisogno di installare nulla sul proprio PC.

Anche per l'Intel 4004 esiste un interessante progetto di emulazione on-line che si trova all'indirizzo <http://e4004.szyc.org/>.

In poche pagine dal layout minimalista troviamo tre elementi che fanno parte di questo progetto: l'emulatore, un assembler e un disassembler.

L'emulatore è la parte che ci interessa maggiormente in questo contesto. Si presenta come una serie di box a sfondo nero dove la label in giallo e i valori in bianco (rosso se sono in fase di aggiornamento) consentono di tenere sotto controllo una macchina virtuale formata dai quattro elementi fisici: la CPU, il chip del microcodice,

la RAM e lo shift register/IO chip.

Il programma emula in maniera completa il kit MCS-4 della INTEL, una piastra all-in-one adatta alla valutazione del primo microprocessore progettato e costruito dalla ditta che avrà in seguito un futuro così luminoso.

L'uso dell'emulatore non è proprio "user-friendly" e se si deve inserire un valore in una cella di memoria è necessario attivare la cella e poi attraverso un dialog box (vedi figura sotto), procedere all'inserimento del valore desiderato. Questa scomodità è lo scotto che si paga all'ambiente Javascript, potente quanto si vuole ma certo non include alla massima goduria in termini di interazione utente.

Il progetto è stato pensato da Intel come una sorta di sistema embedded, cioè veniva sviluppato un certo codice che poi inserito nella ROM costituiva la parte programma. La RAM, per un massimo di quattro banchi selezionabili da software grazie ad altrettante linee dedicate sul processore, serve come storage dei dati e come area di lavoro per gli stack ed i registri.

La finestra in basso a destra, indicata come "ROM block" è dove deve essere inserito il codice del programma. Questa operazione si può compiere direttamente sia come codice assembly, oppure caricato da

un file di testo esterno.

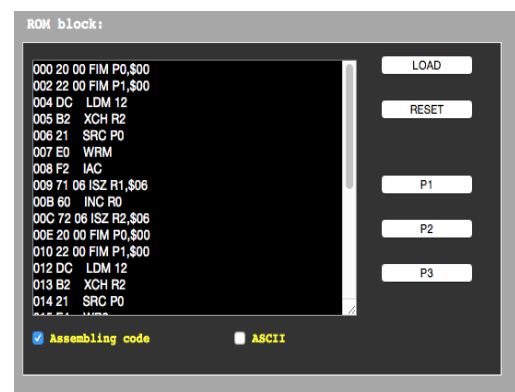
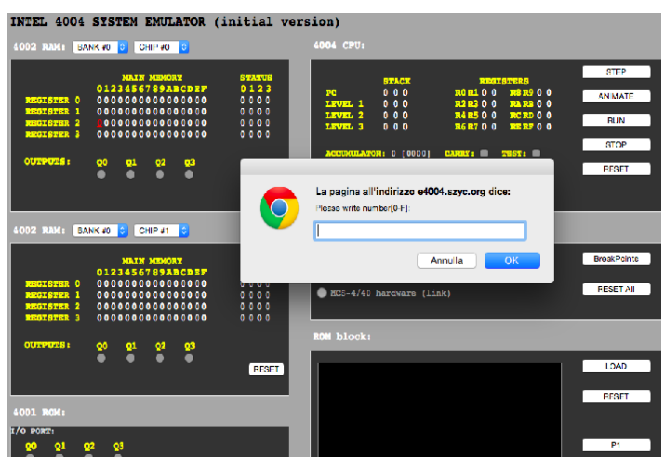
L'esecuzione può avvenire anche step-by-step, come si conviene per un vero emulatore che debba servire alla messa a punto del codice del programma.

### Conclusione.

Programmare il sistema 4004 è tutt'altro che una passeggiata! E' necessario fare i conti con la poca disponibilità hardware e con la scarsità di istruzioni, oltre che con un assembler che non rientra nella categoria dei macro-assembler.

La logica sulla quale sono stati impostati i progetti successivi, a cominciare dal mp 8008 della stessa Intel, ha certamente tenuto conto delle difficoltà del progetto e reso la vita del programmatore molto più agevole e produttiva.

(\*)



# I primi 20 anni della PlayStation



di Lorenzo/2

Correva l'anno 1994 quando in Giappone la Sony faceva uscire la prima versione della PlayStation (quella che oggi viene individuata come PlayStation 1). L'anno successivo toccò a Stati Uniti, Australia e via via agli altri paesi industrializzati fino a raggiungere la ragguardevole cifra di 102 milioni di unità vendute nei dieci anni di commercializzazione.

## **Contesto storico.**

Subito dopo il 1990 il comparto dei giochi vedeva due filoni principali: il PC e i sistemi dedicati venduti soprattutto in Giappone con i prodotti di punta di Nintendo: il Famicom (noto anche come Super Nintendo Entertainment System) e Sega Genesis (noto anche come Mega Drive).

La Nintendo aveva però un problema: doveva espandere il sistema per farlo funzionare con il CD-ROM piuttosto che con le cartridge-ROM che fino a quel momento avevano dettato legge nel mercato delle

console. Il problema era ovviamente quello di allinearsi come prezzi al PC che con il suo CD-ROM nativo poteva permettersi di far pagare meno i giochi visto che il costo del supporto era praticamente nullo mentre una cartuccia di espansione un minimo di hardware e di confezionamento lo richiedeva.

Dato che Nintendo non aveva la tecnologia pensò di allearsi con Sony, che invece di CD-ROM se ne intendeva e molto, per realizzare una espansione del Famicom che permettesse a quest'ultimo l'utilizzo del supporto ottico. A Sony la cosa piaceva perché direttamente produttrice di drive e venderne qualche altro milione certo la attirava.

Questo progetto abortì nel 1988 per incomprensioni sia commerciali che tecniche fra le due aziende. dal punto di vista strategico forse l'errore che minò l'idea era quello dell'espansione invece che puntare su un progetto che avesse il CD-ROM come master di memoria. Un tentativo di andare

nella direzione della console CD-ROM master ci fu ma nel 1988 fallì definitivamente. A questo punto Sony, che non era più legata ad un contratto commerciale con Nintendo, portò avanti da sola il progetto e fece nascere la PlayStation sotto la guida di Ken Kutaragi che poi sarà chiamato “il papà della PlayStation”.

Nintendo cercò un partner sostitutivo che credette di aver individuato in Philips, ma l'accoppiata non funzionò e quella che doveva essere la nuova console di Nintendo non vide mai la luce. Qualche maligno sostiene che non fu colpa né di Sony né di Philips ma era proprio la Nintendo che non funzionava nella collaborazione!

### La PlayStation 1.

Il lancio in Giappone del nuovo prodotto per l'intrattenimento domestico avvenne il 3 Dicembre 1994, giusto in tempo per Natale, mentre il resto del mondo dovette aspettare quasi un anno dopo il lancio per utilizzare questo oggetto.

Molto del successo di vendita fu dovuto alla scelta indovinata di titoli pronti in esclusiva o ridotti da altre piattaforme che portavano la PS un bel po' avanti rispetto alle console concorrenti.

Ecco l'elenco dei 10 giochi più venduti con relativa quantità (in milioni di unità):

1. Gran Turismo – 10.9
2. Final Fantasy VII - 9.7
3. Gran Turismo 2 – 9.4
4. Final Fantasy VIII – 7.8
5. Cortex Strikes Back – 7.5
6. Tekken 3 – 7.1
7. Crash Bandicoot 3: Warped – 7.1
8. Crash Bandicoot – 6.8
9. Driver - 6.27
10. Metal Gear Solid - 6.03

### La PlayStation 2.



L'anno 2000 vede la competizione fra due nuove piattaforme che promettono miracoli: il Sega Dreamcast e la PlayStation 2 (PS2). Il primo giorno di lancio il Dreamcast aveva battuto un record incredibile: 97 milioni di dollari di vendite! Sembrava impossibile ma la PS2 fece di meglio e non poco: 250 milioni di dollari nelle prime 24 ore. L'incredibile successo di vendita ebbe un risvolto negativo: mise in crisi la produzione che non riuscì per settimane a far fronte agli ordini che arrivavano.

Dal punto di vista tecnico la PS2 rappresenta un salto tecnologico enorme rispetto al sistema che la precede a cominciare dal supporto ottico che passa dal classico CD-ROM a 640 Mb al più moderno DVD-ROM dual layer fino a 9 Gb.

La PS2 è un sistema di calcolo molto più

avanzato della PS1 che pure era una macchina di tutto rispetto all'epoca del lancio con la sua CPU a 32 bit a 33 MHz. Il processore che equipaggia la PS2 è a 64 bit con clock a 299 MHz PS2 e la scheda video capace di una risoluzione standard di 720x480 pixel. Chi sa leggere cosa sta dietro a questi numeri capisce come fra le due macchine ci sia un solco profondo, una voragine addirittura!

Per la prima volta una console da gioco si poteva collegare on-line e quindi ospitare delle partite virtuali fra più giocatori; un altro passo nella direzione della globalità.

Fra gli oltre 3700 titoli rilasciati vanno ricordati i best seller seguenti:

1. *Grand Theft Auto: San Andreas*
2. *Grand Theft Auto: Vice City*
3. *Gran Turismo 3: A-Spec*
4. *Grand Theft Auto III*
5. *Gran Turismo 4*
6. *Final Fantasy X*
7. *Need for Speed Underground*
8. *Need for Speed Underground 2*
9. *Medal of Honor: Frontline*
10. *Kingdom Hearts*

Il campione di vendite è proprio *San Andreas* con le sue oltre 20 milioni di copie vendute! Si calcola che globalmente il numero di copie di giochi venduti per questa console assommi a 1,5 miliardi!

Quante PS2 si sono vendute? Anche qui il numero fa impressione perché a tutt'oggi è il sistema che ha venduto più unità in assoluto: 155 milioni di utenti ne hanno comprata una!

### **Play Station Portable.**



E arriviamo nel 2004, in piena epoca PS2, quando Sony pensò di racimolare quel segmento di mercato che era stata appannaggio della concorrenza e nella fatti specie dal Game Boy nelle varie versioni. Cioè il comparto dei dispositivi di gioco trasportabili (handheld console, come sono chiamate).

Nasce così la PSP (Play Station Portable) che cerca di scalzare l'aguerrita concorrenza della Nintendo con un sistema dalle capacità grafiche stratosferiche (per l'epoca). Oltre alla risoluzione video ci voleva un sistema di storage capiente e Sony scelse una versione ridotta del supporto ottico stile cd-rom in un case protettivo di plastica. Questo supporto è chiamato Universal Media Disc.



Purtroppo per Sony la prima versione del-

la PSP “godeva” di qualche difetto; ad esempio espelleva il disco se si orientava improvvisamente secondo un certo angolo. Così le vendite non sono state ai livelli che l'azienda si aspettava pur con il rilascio di varie versioni con capacità multimediali, cioè in grado di visualizzare dei film e di collegamento on-line per il gioco multiutente o la navigazione in Internet.

Il gioco che ha venduto di più è *Grand Theft Auto: Liberty City Stories* con le oltre 7 milioni di copie ma ovviamente molto lontano dai grandi numeri delle sorelle maggiori PS1 e PS2.

### **Un nuovo concorrente.**

Il successo della PS2 e l'insuccesso contemporaneo dei giochi per Sega Dreamcast (alcuni giochi di sport furono proprio sbagliati per il gusto occidentale), affossarono la Sega come azienda di un certo peso nel comparto dei videogame. Rimase la Nintendo ma nel 2001 cominciò l'avventura della Microsoft con il suo Xbox seguito nel 2005 dalla versione 360, vero concorrente di Sony per l'intrattenimento domestico.

Per non perdere la leadership la Sony fu costretta a correre per tenere distante la temibile concorrente. Nel 2006 venne rilasciata la PS3, degna concorrente della PS2 e rivale acerrima della console di Microsoft.

### **La PlayStation 3.**



Sul finire del 2006, al solito in tempo per i regali di Natale, la PS3 si presenta con tecnologie nuove e per certi versi sorprendenti. Prima di tutto utilizza come supporto addirittura un blu-ray e contemporaneamente crea il proprio network di gioco chiamato PlayStation Network (PSN) in diretta concorrenza con il servizio corrispondente di Microsoft che si chiama Xbox LIVE.

Qualcuno sostiene che la Sony vendesse in netta perdita le prime tranches del suo nuovo prodotto per effetto del costo del drive Blu-Ray. Comunque alla fine del 2013 quando venne soppiantata dalle PS4, la PS3 aveva venduto oltre 80 milioni di unità e senza dubbio la Sony aveva coperto le spese e ci aveva guadagnato qualche bel soldino!

Uno dei titoli “mitici” di questa console è *Call of Duty 3* anche se *Gran Turismo* con le sue 10 milioni di copie vendute rimane il campione di vendite.

Della PS3 fu molto criticato il Joystick a forma di boomerang, presto sostituito con la forma più tradizionale vista per la PS2.

### **La PS-VITA.**

Nel 2012 la Sony ha tentato la rivitalizzazione del dispositivo di gioco portatile con la Playstation VITA. Questo dispositivo rimpiazza la PSP ed è tutt'ora in listino. Alla fine del 2013 aveva venduto circa quattro milioni di esemplari. Il mercato delle handheld game console rimane un mercato di nicchia dove però evidentemente "bisogna esserci".

La PS-VITA ha un display più grande della PSP che è anche un dispositivo touch così come porta sul retro un touchpad. Ingloba la connettività più varia (ma anche la PSP aveva il WI-FI, peraltro in modalità WEP poco sicura); contiene un giroscopio a sei assi e una bussola tridimensionale. La Sony è riuscita a organizzare ben 16 bottoni in questo piccolo dispositivo e due joystick analogici "a ruota".

La PS-VITA fa della connettività il suo punto di forza e può essere usata anche come console remota compatibile con alcuni giochi della PS3.

La tendenza, mantenuta anche nel prodotto "master" successivo, cioè la PS4, è quella di rendere sempre più "social" l'esperienza videoludica e costituire inoltre un ambiente coerente fra i vari prodotti: una specie di "universo" insomma.



### **La PlayStation 4.**

Rispettando il periodo tradizionale per il lancio delle console, la Sony nel Novembre 2013 presenta il suo prodotto PS4 proprio un giorno prima della presentazione da parte di Microsoft di quel XBOX ONE concepito per mettere in difficoltà la Sony sul fronte tecnologico, il che si traduce in soldoni nel disporre di un sistema con grafica sempre più spinta ed immersiva.

Immediatamente fu chiaro a tutti che i player rimasti erano solo due: Sony e Microsoft e che le altre piattaforme si sarebbero dovute accontentare di una nicchia sempre più ridotta del mercato.

A distanza di un anno dal lancio la PS4 è leader di mercato con il 44% di share e quasi 14 milioni di unità vendute.

I dati tecnici dell'ultimo prodotto di Sony sono impressionanti: la CPU (o meglio la GPU) è capace di 1843 teraflops (un 50% più veloce della XBOX ONE) rilegando il concorrente a comprimario visto che lo sovrasta in risoluzione e velocità di elaborazione.

La socialità del videogioco viene perseguita e non solo con il multiplayer ma anche con la condivisione degli elementi di gioco e delle partite oltre che dei dispositivi. Con la PS4 esce anche un nuovo controller "atti-



vo": il DualShock 4.

Per quanto riguarda i titoli più venduti si evince la tendenza ad un certo appiattimento: giochi sportivi e in particolare FIFA 15 con i suoi 4 milioni di copie vendute e tutta la gamma "Call of Duty".



### **... e per il futuro?**

L'aspetto "social" dell'esperienza videoludica si è rafforzato ulteriormente nel 2014 con il rilascio della release 2.0 di aggiornamento del sistema. Ora è possibile condividere le partite anche con possessori di PS3 e la PS-VITA diventa una console "embedded" del sistema di gioco.

Si parla di una sorta di cloud ludico dove si eseguiranno i giochi senza dover disporre del supporto fisico e quindi con ancora migliori possibilità di sharing della propria esperienza.

La tendenza generale dei servizi Internet sta andando abbastanza velocemente nella direzione dello streaming. TV, film e ora anche giochi richiederanno sempre meno apparecchi specializzati e sempre di più banda per essere usufruiti.

Questo da un certo punto di vista non può che inquietare noi italiani, costretti per ragioni politiche e lobbies di cartello alla vecchia e poco scalabile ormai ADSL su cavo di rame. Speriamo che ci si schiodi da questa visione retrograda e si capisca che il futuro non sta nel 100 Mbit/s, fra l'altro poco realistici sul doppino di rame, ma nel giga come minimo e fra non pochi anni magari nei 3/4 Gigabyte/s direttamente sul router di casa.

Ho paura che vivremo un nuovo digital-divide e questa volta ad essere tagliati fuori non saranno i piccoli paesini "poco profittevoli per il business", come i provider attuali citano, ma l'intero Paese.

Chi vivrà... giocherà (forse).

(\*)

