

1 Lezione del 03-03-25

1.1 Video

Il supporto principale al video e' la **memoria video**, che lato software si comporta perlopiu' come una normale memoria ad accesso casuale.

Questo e' quindi il primo esempio di un oggetto che si trova nello spazio di memoria, senza necessariamente *essere* memoria: cio' che vi viene scritto non viene memorizzato, ma visualizzato sullo schermo.

Inoltre, la memoria video supporta un accesso *bidirezionale*: cioe' vi si puo' accedere sia lato CPU che lato **adattatore video**, cioe' la rete che si occupa di gestire tale memoria e visualizzarla sul *display*. Lo standard appunti lett prevede che l'adattatore sia configurabile e utilizzabile in due modalita':

- **Modalita' testo:** ogni locazione viene associata ad un carattere ASCII da visualizzare sullo schermo, diviso in 80 colonne \times 25 righe. E' questa la modalita' di default in cui si avvia l'adattatore.

In questo caso il compito dell'adattatore e' quello di leggere i quanti KB di memoria, e convertire ogni codice nel carattere principale. Questo viene fatto consultando una ROM di caratteri che contiene quello che e' effettivamente il *font* dell'adattatore. Solitamente si puo' anche redirezionare la lettura in ROM ad una certa regione della RAM, modificando cosi' il font.

La faccenda e' veramente piu' complicata: si dedicano non 1 ma 2 byte ad ogni carattere, dove il byte piu' significativo rappresenta informazioni riguardo al **colore** del carattere:

- I 4 bit meno significativi rappresentano il colore del *foreground*;
- I 3 bit successivi rappresentano il colore del *background*;
- Il bit piu' significativo rappresenta il *blinking*, cioe' indica all'adattatore di far lampeggiare quel carattere nel tempo.

La modalita' testo non ha idea della posizione del cursore sullo scherm: attraverso registri si puo' indicare la posizione del cursore, e modificando la regione di memoria interessata si possono cambiare i caratteri in qualsiasi zona dello schermo. Il comportamento del cursore (spostamento, ritorno a capo, ritorno carrello, ecc...) e' quindi gestito interamente lato software.

- **Modalita' grafica:** programmando i registri dell'adattatore si possono ottenere diverse modalita' grafiche, che permettono al programmatore di colorare singoli pixel sul display. Nella macchina virtuale usata incapsuliamo tale operazione di conversione in un apposita libreria, e scriviamo pixel con colori su 8 bit (per 256 colori diversi). Nei sistemi moderni la memoria video non viene scritta dalla CPU, ma da un *coprocessore grafico* che esegue un suo programma, mentre la CPU puo' dedicarsi ad altro.

1.1.1 Indirizzamento dei registri dell'adattatore video

Vediamo nel dettaglio come si possono indirizzare i registri interni dell'adattatore video. Questo dispone infatti di una vasta gamma di registri, ma una sola linea di ingresso da un byte per indirizzamento e scrittura. Le scritture vengono quindi eseguite in serie:

- Prima specificando l'**indirizzo** del registro da aggiornare;
- Poi inserendo i **dati** da scrivere a tale indirizzo.

1.2 Timer

Il timer e' realizzato come un interfaccia ad eventi, che riceve in ingresso un clock e aggiorna ciclicamente un registro contatore. Al raggiungimento di 0 da parte del contatore, si resetta e si invia un certo evento (un impulso).

Nel PC IBM in particolare troviamo 3 contatori:

- **Contatore 0:** e' collegata a mistero
- **Contatore 1:** era storicamente usato per il refresh della RAM, oggi non viene piu' usato;
- **Contatore 2:** era collegato all'unico dispositivo audio presente sull'IBM, cioe' il beeper speaker.

1.2.1 Sonoro

Vediamo in particolare il lato sonoro del PC IBM. Essendo stato questo un calcolatore pensato per l'uso da ufficio, le capacita' audio erano molto limitata: si disponeva di un beeper speaker a frequenza modulabile dal timer (contatore 2). Inoltre, un particolare registro in memoria era collegato direttamente in AND con l'uscita del contatore 2, permettendo la modulazione on/off del segnale allo speaker.

Qesto tipo di modulazione permetteva effettivamente di sfruttare, in maniera non prevista dalla IBM, per riprodurre segnali generici.