

Un videogioco tutto nostro

Siamo giunti al nostro penultimo resoconto sullo stato del nostro videogioco. Il prossimo mese vi dovrei presentare il programma finale e allora tireremo le nostre belle conclusioni. Per questa puntata ci occuperemo di osservare, come al solito, la situazione raggiunta

In quest'ultimo mese ho realizzato gran parte del lavoro «concreto», vale a dire programma già testato e funzionante e totale della grafica, più alcuni esperimenti musicali. Chiaramente il programma non è ancora finito, altrimenti potevamo concludere con questa puntata, e manca la colonna sonora (ovvero c'è il programma per realizzarla ma non il musicista... ovvero c'è il musicista ma non l'idea per la musica... ovvero c'è l'idea ma non il tempo per realizzarla... fino ad oggi). Ci incamminiamo quindi in una serie di spiegazioni che, spero, serviranno per far rivivere anche a voi le mie stesse emozioni riguardo questo progetto.

Al termine della scorsa puntata mi ritrovavo con alcune schermate realizzate tramite Kwik paint (un prg per disegnare sul Commodore 64) e un grande schema della struttura del gioco, più una manciata di routine per la presentazione tra l'altro neanche testate «bene». L'unica parte del programma che era stata effettivamente trasferita sul computer e vista funzionare era la routine musicale. Le schermate erano relative alla presentazione, ma in passato ne avevo realizzate diverse come modelli per il gioco effettivo e molto spesso erano state scartate perché il gioco si era modificato lungo il cammino. Nonostante ciò mi era piaciuta l'idea di alcune schermate pubblicate sul numero di gennaio dell'88 (più di un anno fa!). Ovviamente non andavano bene per il

Megagame attuale, ma il loro stile mi convinceva. Vada per lo stile, mi metto quindi all'opera per adattare ai pannelli rimodernati (vedi numero di novembre '89). E in queste pagine potete infatti ammirare i nuovi pannelli made in C-64.

Terminate le schermate con il Kwik paint occorre renderle utilizzabili, quindi trasformarle in set di caratteri. Questo era solo il primo passo da fare in quanto, una volta mescolate nei set (tre) di caratteri occorrevo delle mappe che avrebbero ricostruito le schermate secondo i nostri comandi. La soluzione era più semplice del previsto in quanto mi occorreva un programma in grado di «costruire» queste mappe e a suo tempo avevo realizzato qualcosa del genere per il videogame Psicoman (rimasto incompleto...); bastava modificare «un po'» di cosette. Mi ritrovo alla fine con un bel tool per la creazione delle mie mappe, ma mi servivano ancora i set di caratteri. Il primo dei tre riguardava la grafica della presentazione (stelle, montagne, scritte e set alfabetico) che ha dato fortunatamente pochissimi problemi (solo un carattere non coincideva con una scritta). Il procedimento di trasformazione consisteva nel «ritagliare» parti di screen effettivamente occupate dal disegno e comprimerle in una ristretta area di schermo in hi-res; fatto questo bastava salvare la schermata e poi prelevare solo la parte occupata (l'inizio della screen... i primi 2 kbyte, coincidenti appunto con l'ampiezza di un set di caratteri). Come forse non sapete, o non ve ne siete mai resi conto, le screen in hi-res (sia multi che mono color) sono memorizzate in ram con lo stesso ordine della mappa caratteri, ovvero non c'è bisogno di conversione tra i due formati (a differenza degli sprite), quindi il «gioco» era già fatto senza ulteriore sforzo; tutto stava a spostare la parte di schermata che ci interessava nell'area ram da noi dedicata al (primo) set di caratteri. Chiaramente la procedura dei «tagli» illustrata prima richiede una certa attenzione in quanto dobbiamo far coincidere esattamente in un'area di 8 per 8 pixel (allineata con 8 byte) ogni carattere, sempre che questo debba essere utilizzato da solo come nel caso dei caratteri facenti parte del set alfabetico; nel caso di



gruppi di caratteri che poi saranno stampati uno accanto all'altro non ci sono particolari problemi di sconfinamento, almeno per quelli interni al gruppo. Tutto questo per farvi capire che mi sono ritrovato più volte a dover riaggiustare la schermata «tagliuzzata» perché un determinato carattere di 8 per 8 era diviso in due, oppure non si incastrava bene nel mosaico (specialmente per quanto riguarda il set dei pannelli). Il Kwik paint utilizza solo il modo multicolor e infatti tutto il megagame è in multicolor, caratteri alfabetici compresi. Per gli sprite, che poi si riducono ai soli cursori animati, ho usato un programma di conversione già utilizzato per lo stesso Psicoman. Tanto per la cronaca il secondo set è quello dei pannelli e il terzo quello della schermata del «famoso» fine-giorno. Nel caso qualcuno di voi volesse impossessarsi del mio tool per la creazione di mappe caratteri vi illustro alcune sue caratteristiche. Sullo schermo è visualizzata una bella fetta di quella che è la nostra pagina video, più l'intero set di

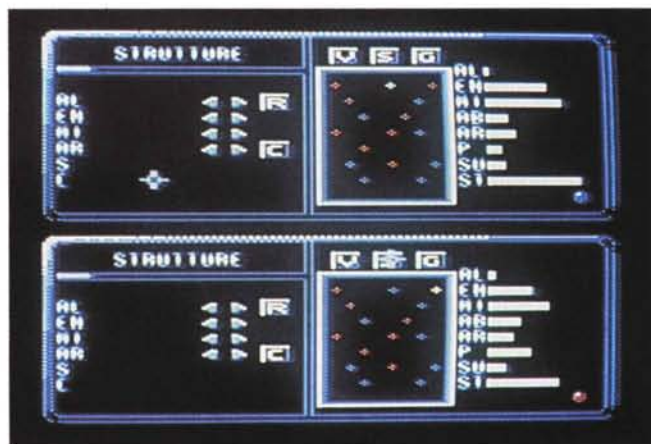
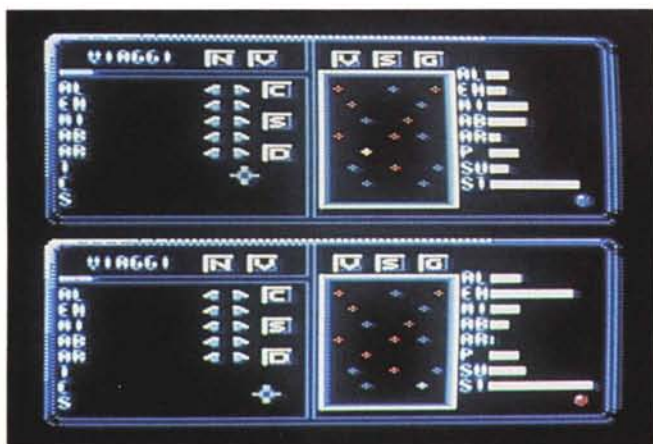
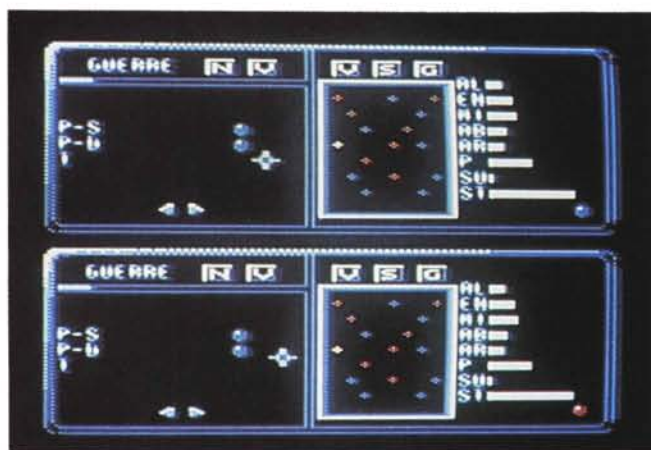
caratteri (sia ridefiniti che normali, a scelta). Si costruisce tramite joystick con il cursore che si trasforma nel carattere da noi scelto tra i disponibili. Possono essere definiti fino a 9 «pennelli», ovvero agglomerati di caratteri, prelevati dallo screen stesso o dal set, richiamabili in qualunque momento (con notevole risparmio di tempo). Tramite altre opzioni è possibile richiamare screen memorizzate in ram secondo un ordine scelto da noi (in quanto è possibile riposizionare liberamente il puntatore che stabilisce la sezione di ram) e ovviamente salvarle (sempre in ram). C'è anche la gestione dei file su disco (save e load). Se vi interessa fatemelo sapere che ne riparliamo. Torniamo al Megagame. Siamo giunti al momento in cui mi ritrovo finalmente con le mie mappe e i miei set di caratteri, più gli sprite. Stabilisco la loro posizione in ram e li salvo in un unico blocco, al fine di poterli ricaricare ogni volta che effettuerò un test sulle routine Assembler che mi accingo a realizzare.

Le prime routine che trasferisco sul Commodore, servendomi del solito Assembler, sono quelle della presentazione. Mi accorgo che devo fare alcune modifiche perché parte di esse possono essere utilizzate come sub anche per il resto del programma e perché ci sono alcune imperfezioni. Dopo non poche lotte con la routine dello scrolling parallaxico, che non voleva assolutamente essere fluido (ma l'ho spuntata io) la presentazione è completa e la mostro orgogliosamente a mio fratello. Mancano ancora i nomi da inserire nella lista degli «hanno partecipato». Ma la routine che più mi rende felice è quella della stampa di fette di ram di qualunque dimensione, affiancata poi da quella di riempimento con valore fisso (per le cancellazioni). L'unico inconveniente è che per funzionare a volte richiedono (se la fetta è troppo grande) più di un ciclo raster (1/50 di secondo), ma ciò non crea alcun problema (almeno fino ad oggi!).

Quello di cui ho bisogno a questo punto è un elenco dettagliato delle loca-



Sarei tentato a lasciarla così.



zioni di partenza di tutte le schermate contenute nel blocco dati della mappa realizzata in precedenza. Mi realizzo una modifica al programma di editing delle mappe in modo che sia lui a fornirmi le mie informazioni e il problema non sussiste più. In questo elenco occorre aggiungere anche le locazioni dello schermo che andranno ad occupare, una volta stampate le relative screen, i tasti dei pannelli, le scritte seguite da informazioni a barra, la posizione dei pianeti, più gli indicatori vari. Anche se il tutto è diverso nel caso dei pannelli del secondo giocatore, basta aggiungere un delta di 520 posizioni ed i dati ottenuti risultano perfettamente coincidenti.

Consideratelo pure un capriccio da programmatore, ma le prossime routine che mi propongo di realizzare sono quelle di gestione dei joystick, della stampa dei pianeti sulle mappe e della stampa degli indicatori a barra. Butto giù degli schizzi e passo subito alla scrittura delle routine in Assembler; in effetti l'impresa mi era sembrata più complessa di quanto effettivamente poteva essere dopo alcune semplici modifiche di criterio. Durante l'assemblaggio di tanto in tanto carico il blocco di dati con la grafica per assicurarmi che tutto vada per il meglio, ma il più delle volte è solo perché sono curioso di vedere il risultato di ogni minima modifica (e la cosa mi comporta un bel po' di carichi-attese-perditempo). Tanto per avere in mano qualcosa di concreto mi faccio un paio di demo con il risultato di quanto ho fatto fino a quel momento.

Andiamo ad esaminarci queste routine. Cominciamo con quella di gestione dei joystick. Come abbiamo detto il cursore-puntatore è uno sprite (animato) quindi le sue coordinate non sono direttamente utilizzabili per essere confrontate con la posizione sul video dei vari gadget, che sono invece in formato 40x25. Tuttavia è sufficiente dividere per 8 sia la coordinata X che la Y dello sprite e otteniamo una coppia compatibile con lo schermo caratteri. Il carattere in alto a sinistra (locazione ram-video 1024) corrisponde alle coordinate 0,0 (X,Y) e il secondo (loc. ram 1025) alle coordinate 1,0 e così via. C'è ancora un piccolo problema da risolvere, in quanto quando lo sprite occupa le coordinate 0,0 non coincide (in termini di sovrapposizione fisica) al primo carattere e per questo occorre aggiungere una delta di 22 pixel in X e 50 in Y (considerando anche la forma dello sprite). Tutto ciò che riguarda questa routine funziona in IRQ video.

Passiamo alla routine di gestione della stampa degli indicatori a barra. Questi indicatori sono costituiti da una barra di

pixel in orizzontale (avente una ampiezza verticale di 4 pixel), tanto più ampia quanto maggiore è la grandezza che con essa si intende rappresentare. La barra è quindi costituita da un insieme di caratteri ridefiniti in multicolor. La sua max ampiezza orizzontale è sempre in 54 pixel (quindi 8 caratteri). Se osservate le foto relative ai pannelli potete ritrovare una serie di 8 barre nella destra della mappa. Il mini set di caratteri abbinato alla routine è costituito da 8 «pezzi», aventi ognuno una striscetta di quattro pixel verticali in più rispetto al precedente. Per realizzare una barra di 10 pixel occorrono 2 caratteri, uno di 8 pixel (pieno) e l'altro di 2, messi uno accanto all'altro. La routine trasforma un qualsiasi numero compreso tra 0 e 63 in una corrispondente barra, posizionata sullo schermo a nostro piacere. Il suo funzionamento è abbastanza semplice. È sufficiente avere un puntatore alla locazione video di partenza e il numero da trasformare. Del numero si prendono i tre bit meno significativi e si depositano in un registro libero; questi tre bit indicano quale degli 8 caratteri occorre stampare dopo la serie di caratteri «da 8 pixel». Quest'ultima serie di caratteri viene stabilita tramite una divisione per 8 del numero stesso. Si effettua quindi un ciclo di stampa servendosi del puntatore video e incrementandolo di volta in volta. Il ciclo ovviamente dura tanto quanto è il risultato della divisione (nel caso dell'esempio, un solo ciclo). Al termine del ciclo si stampa il carattere stabilito dai tre bit meno significativi, ricavati in precedenza, nella posizione video indicata dal puntatore attuale.

Veniamo alla routine di stampa dei pianeti. La mappa è di 8x7 caratteri, ciò vuol dire che è compresa in un range di $6 \times 40 + 8 = 240$ locazioni, quindi è sufficiente l'indirizzamento indicizzato per ricoprire l'intera area. La loro posizione effettiva sullo schermo è determinata da una locazione di base (diversa, ovviamente per le due mappe, dei due giocatori) più una delta (registro X) che va da 0 a 240, contenuto in una opportuna tabella di 16 elementi. La loro forma dipende dal loro stato (normalità, pericolo o guerra) e per stabilirla occorre ricercare i flag che lo indicano nella tabella dati del pianeta. Anche il colore di stampa dipende da questi dati (mio-tuo, attivo-disattivo).

E adesso occupiamoci della routine che utilizzano come «schiavi» le sub finora descritte, ovvero la routine di gestione dei tasti e conseguente attivazione delle funzioni corrispondenti, la routine di gestione dei pianeti e la routine di gestione della stampa dei pannelli. In parte queste routine si fondono e in parte svolgono funzioni a sé stanti. Ad esempio, la routine dei tasti deve controllare che il joystick coincida con uno dei tasti contenuti nei pannelli ovvero

che le coordinate di questi, contenute nelle tabelle già descritte, coincidano con quelle del puntatore; anche per la gestione dei pianeti occorre un confronto di coordinate per stabilire quale è quello attivo, quindi a quale devono riferirsi le caratteristiche stampate con le barre. La routine dei tasti non fa altro che settare un flag quando verifica la coincidenza; sarà poi il resto del programma ad occuparsi della funzione relativa. Quando si attiva un nuovo pianeta la mappa viene ristampata per aggiornare «l'illuminazione» dei pianeti, ovvero per spegnere quello vecchio e accendere quello nuovo (vi ricordo che un pianeta è acceso quando passa dal suo colore base, blu o rosso, al rispettivo colore più luminoso, ciano o giallo), quindi viene ristampato l'elenco di base sulla destra, prelevando i dati dalla tabella del pianeta attivato.

La stampa dei pannelli si effettua basandosi sulle seguenti informazioni:

— giocatore utilizzatore;

— tipo di pannello;

— tipo di sottopannello;

quindi la prima informazione determina dove stampare il pannello, la seconda qual è l'opzione e la terza qual è la sotto opzione. Si procede quindi con la stampa, solo se i pannelli attivi sono cambiati rispetto al ciclo programma precedente.

Megaposta

Caro Marco Pesce,
finalmente tra tutti i lettori della rivista MC c'è anche una ragazza che s'interessa di videogiochi. Compro regolarmente la rivista e leggo volentieri anche la rubrica di Francesco Carlà, pur trovando spesso discutibili i suoi giudizi.

Ritengo la tua iniziativa veramente meritevole dello spazio che ha occupato finora nella rivista (se non vado errata, sono più di 2 anni che il Megagame «procede»). Mi rendo conto che è stata un'impresa difficile, ma spero che tu possa persistere perché trovo veramente interessante quello che scrivi e penso di non essere la sola, e spero che tu possa continuare ancora per molto tempo. Spero che questa lettera ti sia di conforto in qualche modo, ma non credo che tu ne abbia bisogno vista la tua tenacia. Aspetto con impazienza il prossimo numero!

A presto

Barbara Tosti, Roma

Erano due anni che aspettavo una lettera così. A quanto pare l'attesa è stata più che ricompensata! Cara Barbara, spero proprio che questa non sarà la tua ultima lettera... prossimamente fammi sapere se sei interessata a qualche argomento particolare.

Ciao.



COMPUTER GRAFICA

PROFESSIONALE E SEMIPROFESSIONALE

Fornendo qualificata assistenza tecnica pre e post-vendita

- ✦ **Amiga 500, 1000, 2000**
- ✦ **Espansioni e periferiche con tecnologia avanzata a prezzi competitivi**
- ✦ **Nuove unità centrali Motorola 68020 e 68030 con velocità fino a 25 MHz che migliorano le prestazioni di 4-6 volte**
- ✦ **DIGITALIZZATORI:**
 - FRAMER in tempo reale a colori**
 - VD-AMIGA in tempo reale in B/N**
 - VIDEON 2 a colori con filtratura elettronica**
- ✦ **GENLOCK:**
 - ECR - VIDEOCOMP - NERIKI - MAGNI**
- ✦ **Stampanti a colori a getto d'inchiostro XEROX 4020**
- ✦ **SOFTWARE ORIGINALE per paint, animazioni, titoli, modellazione solida, effetti speciali, editoria elettronica**

INFORMATICA ITALIA S.r.l.

C.so Re Umberto, 128

10128 TORINO Tel. 011/501647