

# Realizzazione di un core grafico con un FPGA a basso costo

Un team di progettisti di Virtual Logic ha creato un engine di visualizzazione grafica basato su un FPGA sfruttando tecniche tipiche degli anni '80: con questo lavoro si è dimostrato che con l'impiego di avanzate tecniche per la progettazione di system-on-chip e di un dispositivo FPGA a media densità, i progetti che prevedono un micro a 8 bit possono essere sviluppati in maniera più semplice, veloce ed economica

**Rob Evans**  
Altium

**L**a continua discesa dei prezzi degli FPGA, abbinata al costante aumento della loro densità, può indurre a pensare che un progetto che preveda l'uso di componenti programmabili ottimizzati dal punto di vista dello sfruttamento delle risorse sia un concetto relegato al passato, alla medesima stregua dei sistemi operativi completi per PC integrati in due soli floppy disk. Al giorno d'oggi sono infatti disponibili FPGA veloci e di ampie dimensioni che rappresentano la piattaforma ideale per implementare sistemi digitali completi su un singolo dispositivo: nel caso anche questi ultimi dispongano di risorse limitate, i progettisti si trovano a dover affrontare inevitabilmente ostacoli di non poco conto sul loro cammino. Per superare questo tipo di limitazioni i progettisti sono costretti a escogitare differenti e spesso innovativi approcci per il design di un sistema, se non addirittura a ricorrere ad alcuni principi di progettazione del passato. L'individuazione dei limiti delle risorse di un sistema non è certamente un fenomeno nuovo e, una volta reperiti i tool adatti, i progettisti di sistema hanno sempre saputo ottenere il massimo dalle risorse disponibili.

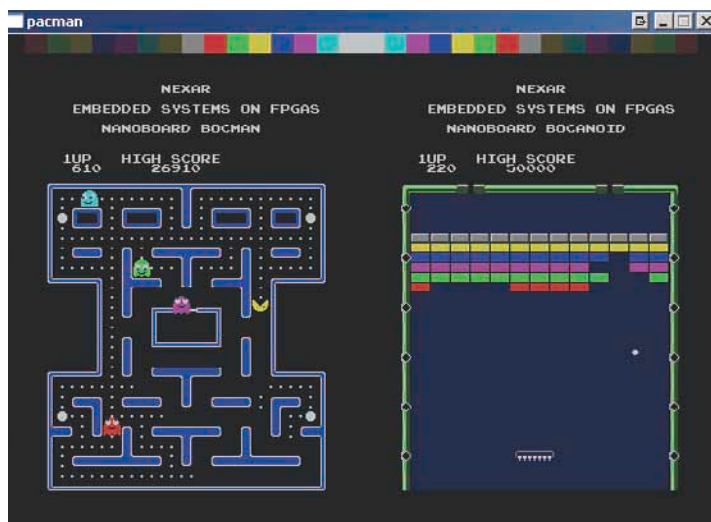


**Fig. 1 - L'impiego della suite di progettazione Nexar e della piattaforma di sviluppo NanoBoard ha permesso a un team di progettisti di Virtual Logic di ricreare i classici videogiochi degli anni '80 sfruttando un singolo FPGA a basso costo**

Per dimostrare che gli insegnamenti del passato non sono andati perduti, un team di progettisti di Virtual Logic ha creato un engine di visualizzazione grafica basato su un FPGA di piccole dimensioni e a elevato grado di configurabilità sfruttando tecniche tipiche degli anni '80. Questo core grafico è il nucleo principale di un emulatore per i vecchi videogame in auge nelle sale giochi degli anni '80 e '90 ed è stato progettato per essere implementato in un FPGA low cost senza ricorrere a periferiche attive come ad esempio a una Ram esterna.

I progettisti hanno scelto l'emulatore di video giochi in quanto rappresenta la piattaforma ideale per lo sviluppo e la dimostrazione delle potenzialità del loro engine grafico. Grazie alla possibilità di integrare un intero sistema basato su microprocessore su un chip low cost, un sistema hardware e software basato su FPGA rappresenta l'ambiente più idoneo per riprodurre i videogiochi classici in auge negli anni '80. Per tutti coloro che hanno familiarità con il progetto MAME (Multiple Arcade Machine Emulator – un programma che permette al computer di funzionare come le vecchie macchine da sala giochi), l'emulatore implementato tramite FPGA può essere considerato un equivalente di tipo embedded. Lo sviluppo dell'emulatore dei videogiochi classici ha anche messo in luce la possibilità di implementare su FPGA sistemi ormai consolidati basati su processori. Progetti come ad esempio quelli relativi a controllori di processo industriali basati su micro a 8 bit e ad altri sistemi che utilizzano processori legacy, potrebbero essere re-ingegnerizzati sfruttando hardware più moderno, efficiente e affidabile utilizzando soft processor più veloci implementati sotto forma di core IP. Il nuovo progetto sarà caratterizzato da innumerevoli vantaggi tra cui ottimizzazione degli ingombri e possibilità di aggiornamento sul campo delle componenti hardware e software, oltre a rappresentare una sorta di archivio storico per il dispositivo originale.

Oltre allo scopo di integrare l'elettronica di una console per videogiochi in un singolo FPGA, il progetto dell'emulatore rappresenta il mezzo ideale per lo sviluppo dell'engine del display sotto forma di un'unità compatta e integrale. Un sistema di questo tipo potrebbe essere successivamente integrato in progetti che prevedono l'uso di FPGA di maggiori dimensioni e richiedono un controllo più complesso della grafica del display o delle animazioni, garantendo in ogni caso un impatto minimo sulle risorse della memoria e del processore.

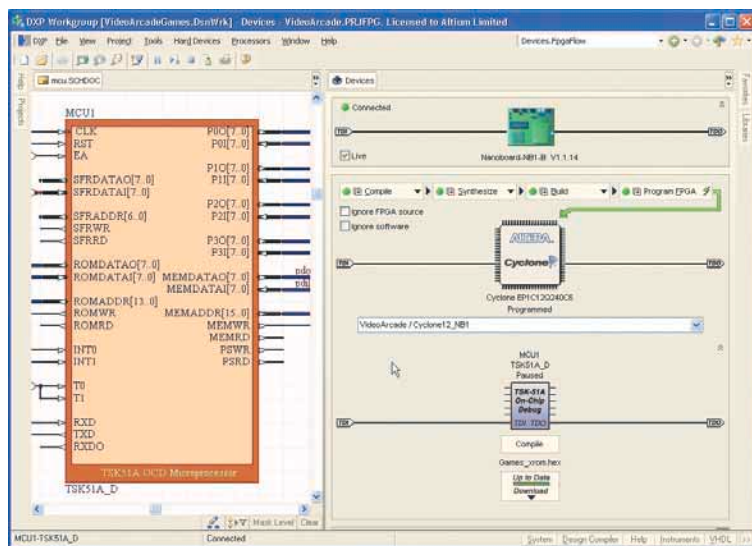


**Fig. 2 - L'uscita sullo schermo VGA prodotta dai videogiochi Bocman e Bocanoid (da sinistra a destra). Per giocare è sufficiente utilizzare un mouse e una tastiera standard, che si collegano direttamente alle porte PS2 della NanoBoard**

### La sfida progettuale

La riproduzione di una tecnologia vecchia di 20 anni in un FPGA che rende disponibili migliaia di porte logiche può sembrare a prima vista un compito relativamente semplice: in questo caso, vista la preponderanza della componente grafica, si presentavano due problemi di una certa complessità. In un progetto tradizionale, la generazione e la manipolazione della grafica sono attività che coinvolgono in maniera preponderante sia la memoria sia il processore a causa della notevole quantità di dati che devono essere elaborati in tempi brevi. Di solito si adotta un approccio di tipo bitmap che prevede l'uso di buffer di frame video di ampie dimensioni per memorizzare, trasferire ed elaborare i dati presenti sullo schermo a livello di pixel. Buffer di questo tipo comportano la disponibilità di risorse a livello di RAM che scarseggiavano nell'FPGA target.

L'intento dichiarato del progetto era invece di integrare l'intero progetto su un FPGA low cost che disponeva di circa 200 kbit di RAM. Per tale motivo non risultava possibile seguire l'approccio tradizionale. Va peraltro sottolineato il fatto che riducendo i requisiti in termini di memoria era possibile realizzare un core grafico integrabile in quasi tutti gli FPGA a bassa densità, rendendo disponibili ulteriori risorse per altri blocchi di sistema, o anche per migliorare le funzionalità grafiche. In apparenza un tale tipo di approccio può sembrare eccessivamente restrittivo: in realtà, riuscendo a "comprimere" l'en-



**Fig. 3 - I dispositivi in logica combinatoria, compresi processori "soft" pronti all'uso, possono essere posizionati e collegati mediante tecniche di schematic entry, mentre i dispositivi hardware fisici e virtuali sono resi possibili grazie a catene JTAG hardware e software: tutte le operazioni vengono svolte in tempo reale**

gine di visualizzazione all'interno di un FPGA era anche possibile eliminare le complicazioni legate alla presenza di una RAM esterna e della relativa interfaccia. Il design complessivo risultava pertanto più semplice e flessibile, richiedeva un numero inferiore di pin di I/O ed era trasferibile senza problemi in altri progetti di sistemi embedded.

La possibilità di utilizzare il core grafico per altre applicazioni rappresentava uno dei principali obiettivi del progetto: un altro scopo era realizzare un engine per display universale che poteva essere riconfigurato in maniera semplice in modo da assolvere ad altri compiti.

### La scelta dei tool

La creazione di una piattaforma per videogame basta su un core grafico flessibile ed efficiente sfruttando le risorse presenti in un FPGA richiede la disponibilità di una piattaforma altrettanto flessibile ed efficiente. Per tale motivo è stato deciso di utilizzare l'opzione Nexar di Altium Designer per il progetto di sistema a livello di FPGA e la scheda di sviluppo per FPGA NanoBoard di Altium.

L'opzione di licenza di Nexar di Altium Designer mette a disposizione un sistema completo per la progettazione di sistemi digitali su una piattaforma FPGA e consente di adotta-

re un approccio di tipo interattivo nella progettazione di sistemi su FPGA. Essa integra progetti HDL e di tipo block-based, componenti basati su FPGA pronti per l'uso, strumentazione virtuale e tutte le funzionalità necessarie per interagire con NanoBoard, così da rendere disponibile una piattaforma per lo sviluppo di sistemi operante in tempo reale.

La capacità di Nexar di implementare le componenti hardware e software del progetto in un medesimo ambiente ha rappresentato un indubbio vantaggio. In primo luogo ha consentito ai progettisti di sviluppare nuovi concetti o idee sull'FPGA e visualizzare istantaneamente i risultati sulla NanoBoard, indipendentemente dal fatto che le variazioni fossero apportate alla parte hardware o a quella software. Questo approccio diretto e in tempo reale – denominato Live Design da Altium – consente ai progettisti di valutare in maniera tangibile gli effetti imputabili a una modifica apportata in fase di design.

Grazie all'ambiente di progettazione Altium Designer Nexar i progettisti di Virtual Logic hanno potuto effettuare una suddivisione delle

componenti hardware e software del design con la massima libertà. Poiché la priorità era rappresentata dall'ottimizzazione dello sfruttamento delle risorse disponibili, risultava vantaggioso poter trasferire una sezione del progetto dal dominio hardware a quello software e viceversa. Oppure risultava anche possibile modificare una componente del progetto a favore dell'altra: ad esempio una semplice variazione del progetto hardware poteva portare a notevoli benefici sotto l'aspetto software.

Dal punto di vista più prettamente pratico, NanoBoard rende anche disponibili tutte le connessioni essenziali per il progetto dell'emulatore di videogiochi, comprese una porta VGA e porte PS2 per il collegamento del mouse e della tastiera. I controllori per l'adattamento delle porte sono disponibili a titolo gratuito come componenti della libreria di componenti IP pre-sintetizzati e preverificati di Nexar, consentendo l'implementazione delle porte sotto forma di un semplice simbolo dello schema circuitale.

Nel corso del design i progettisti sono stati in grado di eseguire il debug e procedere al perfezionamento del sistema in tempo reale con l'ausilio degli strumenti virtuali di Nexar. Integrati nell'FPGA al momento della compilazione, questi strumenti comunicano direttamente con Nexar attraverso una

catena JTAG di tipo "soft" e con l'utente mediante i pannelli frontali virtuali interattivi. Il modulo di I/O a 4 canali si è rivelato particolarmente utile nel corso dello sviluppo dell'engine grafico di base, in quanto il progettista poteva direttamente modificare e monitorare i registri di controllo della parte grafica all'interno dell'FPGA e osservare, in tempo reale, il risultato sullo schermo VGA.

### La componente grafica: tile e sprite

L'impiego dell'ambiente di progettazione Nexar, dei componenti IP disponibili e della piattaforma di sviluppo NanoBoard, ha permesso ai progettisti di realizzare la struttura hardware dell'emulatore di videogiochi utilizzando la tradizionale modalità di schematic entry. I componenti implementati, sfruttando l'acquisizione dello schema erano controllori PS2 (per il supporto della tastiera e del mouse), controllore VGA, blocchi di ROM e RAM (a singola e doppia porta), logica sparsa oltre naturalmente al core del microprocessore a 8 bit. In questo caso il processore adottato è stato TSK51 di Altium, un soft core compatibile a livello di set di istruzioni con la "onnipresente" famiglia di microcontrollori Intel 8031/51 che può essere programmato e aggiornato direttamente attraverso l'interfaccia di Nexar. In questo particolare progetto TSK51 opera a 30 MHz, una velocità di clock di gran lunga superiore rispetto al suo equivalente fisico. Sotto la supervisione del processore, è stato creato l'ambiente grafico utilizzando due sistemi interattivi: uno costruisce l'immagine di background a partire da mattonelle (tile) formate da 8 x 8 pixel mentre l'altro genera gli elementi in movimento dello schermo utilizzando sprite (ovvero rettangolini che si muovono sullo schermo) di dimensioni pari a 16 x 16 pixel. I colori dei pixel sono selezionati da una tavolozza (palette) di 32: in ogni caso i pixel dello sprite dispongono dell'opzione della trasparenza. Questo approccio basato su caratteri, del tutto simile a quello impiegato nei videogiochi degli anni '80, permette di sfruttare in maniera estremamente efficiente le risorse disponibili.

La visualizzazione dello sfondo è realizzato mediante 64 x 60 mattoncini (per un'area totale dello schermo di 512 x 480 pixel) scelti da una matrice di 128 forme di mattoncini programmabile. La forma dei mattoncini e i dati relativi alla mappatura dello schermo vengono immagazzinati nella memoria configurata all'interno dell'FPGA, mentre un controllore della mappa dei mattoncini implementato in HDL sfrutta le coordinate cartesiane X e Y generate dal controllore VGA per calcolare i valori dei pixel dei dati del mattoncino richiesto per la visualizzazione.

Gli sprite, ovvero gli elementi in movimento che compaiono

sul display, possono essere posizionati in qualsiasi punto dello schermo e spostati lungo gli assi orizzontali e verticali. Sono disponibili otto sprite selezionati da una mappa programmabile di 32, con gli sprite attivi caratterizzati da una scala di priorità di sovrapposizione compresa tra 8 e 1 (dove 1 rappresenta la priorità più alta).

Per fornire un elemento necessario alla maggior parte dei videogiochi, l'ottavo sprite, ovvero il giocatore, dispone della funzione di rilevazione della collisione attiva. Questa funzionalità è stata aggiunta più tardi nel processo di sviluppo sotto forma di blocco HDL: i progettisti si sono resi conto che un approccio di tipo hardware risultava essere più efficace rispetto a un'implementazione mediante software, anche se il sistema di sviluppo integrato Altium Designer Nexar/NanoBoard era in grado di supportare entrambi i tipi di approccio.

Grazie all'utilizzo di un approccio prevalentemente hardware all'elaborazione grafica il movimento, il colore e la sovrapposizione degli sprite possono essere controllati attraverso semplici aggiornamenti dei registri. In questo modo il processore viene esonerato da tutto quel complesso di operazioni, oneroso dal punto di vista sia temporale sia dell'elaborazione, altrimenti necessarie per la visualizzazione e gli spostamenti delle componenti grafiche.

L'utilizzo di un simile approccio hardware, di tipo essenziale, alla parte sonora, ha fatto sì che gli effetti sonori dei videogiochi potessero venire memorizzati in una ROM da 4 k e riprodotti attraverso un convertitore A/D sigma delta temporizzato a una frequenza di 4 kHz. L'unico componente esterno all'FPGA richiesto è un filtro passa basso RC.

### Più di un semplice gioco

Per mostrare le potenzialità della piattaforma per videogiochi basata su FPGA, i progettisti di Virtual Logic hanno ideato due moduli software distinti: Bocman, un videogioco controllato mediante una tastiera PS2 e Bocanoid, un videogioco che richiede la presenza di un mouse PS2. Per entrambi i giochi il display utilizzato è il classico monitor per PC, collegato direttamente alla porta VGA della NanoBoard.

Questi videogiochi sono stati particolarmente utili per dimostrare le potenzialità del concetto di grafica basato sui caratteri, anche se tendono a far dimenticare che tutto ciò può essere realizzato mediante un FPGA a basso costo e sfruttando opportuni accorgimenti in fase di ingegnerizzazione. Il nucleo di visualizzazione grafica genera lo sfondo e gli elementi in movimento del display con un impatto minimo sulle risorse temporali del processore e sulla memoria dell'FPGA.

Il team di progetto di Virtual Logic ha anche realizzato una

versione doppia. Poiché il programma relativo a ciascun gioco era stato progettato per girare su un processore TSK51, nella versione doppia i due giochi erano fatti girare come se avessero a disposizione due processori affiancati all'interno di un singolo FPGA. Le risorse in termini di dati audio, tile e sprite sono così stati condivise tra i due programmi.

L'engine di visualizzazione in grado di generare caratteri oggetto di questo design può vantare una vasta gamma di potenziali applicazioni grazie alle dimensioni compatte e alla flessibilità dell'approccio grafico. La capacità di mascherare forme mediante pixel trasparenti rende possibile la realizzazione di elementi standard come ad esempio il puntatore del mouse oppure una clessidra animata, come pure font (serie completa di caratteri).

Con questo progetto si è dimostrato che grazie all'impiego delle più avanzate tecniche per la progettazione di system-on-chip e di un dispositivo FPGA a media densità e basso costo, i tradizionali progetti che prevedono un micro a 8 bit – come i videogiochi e i controllori di processo – possono essere sviluppati in maniera più semplice, veloce ed economica rispetto alle classiche implementazioni hardware discrete. Il sistema di progettazione per FPGA Altium Designer Nexar e la NanoBoard di Altium rappresentano la piattaforma ideale in virtù delle funzionalità di progettazione real time e dell'ampia libreria di componenti IP fornita insieme ad Altium Designer. Si è anche trattato di un sistema utile per comprendere le problematiche legate allo sviluppo di un progetto che disponga di risorse limitate. Le limitazioni in termini di potenza di elaborazione, capacità di memoria e di budget hanno costretto nei decenni passati gli ingegneri a creare soluzioni molto efficienti e innovative per risolvere problemi ingegneristici di una certa complessità.

Non va peraltro dimenticato il fatto che parecchie tecniche di progettazione in auge tanti anni fa sono valide ancora oggi, particolarmente nei casi in cui le funzionalità del dispositivo e le risorse del sistema sono limitate. A causa della progressiva riduzione delle dimensioni e del contemporaneo aumento della complessità dei dispositivi elettronici, l'adozione di approcci innovativi può risultare particolarmente utile per i progettisti, che possono in ogni caso contare sulle possibilità offerte dai più recenti tool EDA. 

**Altium Italia**

readerservice.it n° 26