

Cyrix 6x86

di Andrea de Prisco

A detta della stessa Cyrix, la famiglia di processori 6x86 offre il più alto livello di performance oggi disponibile per i PC desktop. Grazie all'utilizzo di un'architettura innovativa di sesta generazione, i processori 6x86 raggiungono performance da primato.

Ottimizzato per eseguire sia istruzioni a 16 che a 32 bit, il Cyrix 6x86 è pienamente compatibile con il set di istruzioni x86. Raggiunge alte prestazioni attraverso l'utilizzo di due unità intere ottimizzate superpipeline e una FPU integrata. Le unità intera e floating point sono ottimizzate per eseguire un ampio flusso di istruzioni, usando tecniche avanzate che includono il register renaming, l'out-of-order completion, il data dependency removal, il branch prediction e l'esecuzione speculativa. Pur raggiungendo prestazioni superiori già con il software esistente, il Cyrix 6x86 si può offrire performance ancora superiori (con un guadagno pari al 5-10 per cento) utilizzando codice appositamente ricompilato.

La sua architettura super-scalare mette a disposizione due pipeline distinte per eseguire istruzioni multiple in parallelo ed ottenere alte performance. Inoltre, incrementando il numero di stadi delle pipeline si evitano interruzioni nell'esecuzione delle istruzioni ed è offerta alta scalabilità verso le frequenze di clock maggiori. Il meccanismo di Register Renaming permette la memorizzazione temporanea di dati e la loro disponibilità immediata senza attendere che la CPU acceda alla cache interna o alla memoria di sistema

L'architettura

Il 6x86 è il primo di una nuova generazione di processori ad alte performance compatibili x86. Attraverso l'utilizzo di tecniche innovative, il

6x86 elimina molte dipendenze logiche tra i dati e i conflitti di risorsa esistenti in altri microprocessori. Il 6x86 è formato da cinque blocchi funzionali: l'unità intera, la cache, la MMU, la FPU e l'unità interfacciamento col bus.

Gli indirizzi fisici sono calcolati dalla MMU e passati all'unità cache e all'unità di interfacciamento bus (BIU). La BIU provvede all'interfacciamento tra il sistema esterno e le unità interne di esecuzione del processore.

Integer Unit

L'unità intera offre l'esecuzione parallela delle istruzioni utilizzando due pipeline a

sette stadi. Ognuna delle due pipeline, X e Y, può processare molte istruzioni contemporaneamente.

– Lo stadio *Instruction Fetch (IF)* preleva 16 byte di codice dall'unità cache in un singolo ciclo di clock e controlla il flusso di codice per ogni istruzione di salto che può modificare la normale sequenza di programma.

– *Instruction Decode (ID)*. Composto a sua volta da due stadi, ID1 valuta il flusso di codice e determina il numero di byte in ogni istruzione. Fino a due istruzioni per ciclo di clock sono inviate allo stadio ID2.

– *Address Calculation (AC)*. Composto anch'esso da due stadi, AC1 calcola l'indirizzo di memoria per l'istruzione se questa

si riferisce ad un operando in memoria. AC2 esegue ogni funzione richiesta di memory management, gli accessi alla cache o ai registri. Se una istruzione di floating point è individuata, AC2 la invia all'unità FPU per la sua esecuzione

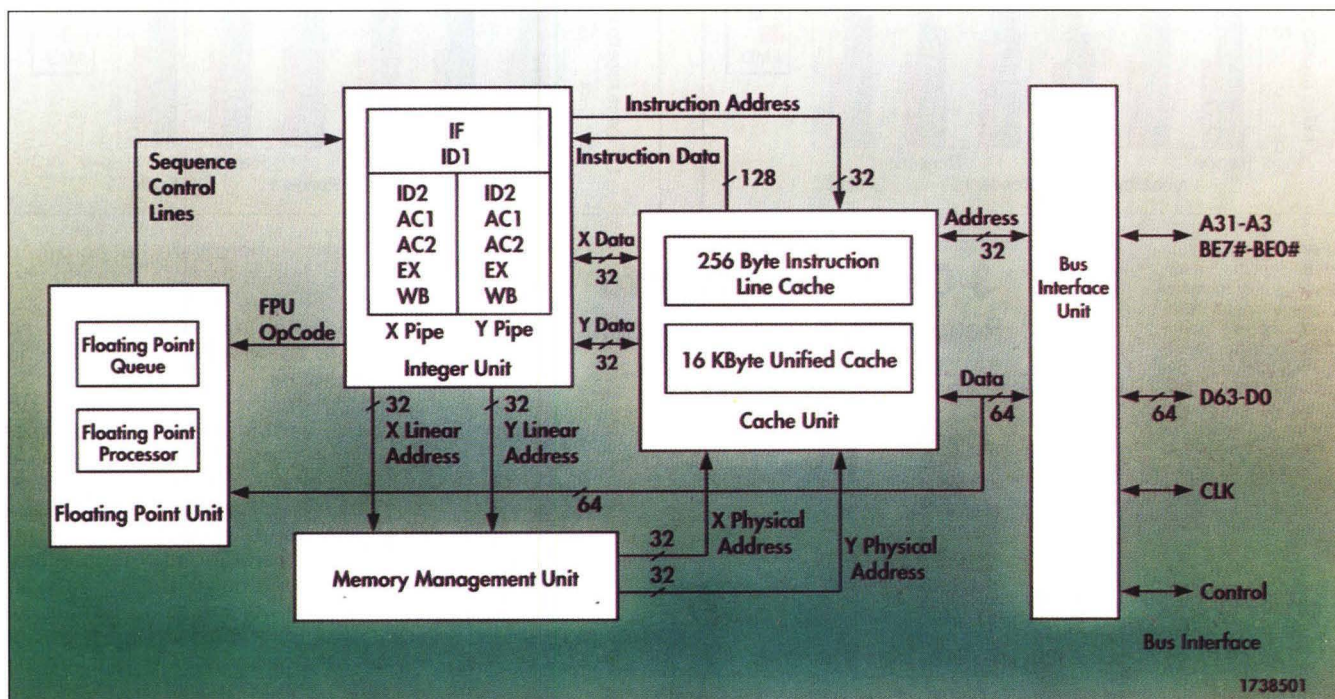
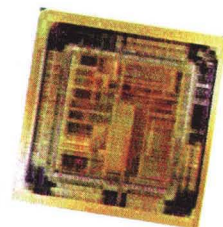
– Lo stadio *Execute (EX)* esegue le istruzioni usando gli operandi messi a disposizione dal precedente stadio AC.

– Lo stadio *Write Back (WB)* memorizza i risultati dell'esecuzione o in un registro all'interno dell'unità intera o nel write buffer dell'unità di controllo della cache.

Out-of-order processing. Se un'istruzione è eseguita più velocemente di una precedente istruzione eseguita nell'altra pipeline, le istruzioni possono essere completate fuori ordine. Il completamento Out-of-order avviene negli stadi EX e WB.

Data dependency solutions. La dipendenza logica dei dati tipicamente forza l'esecuzione serializzata delle istruzioni e può degradare le performance. Il 6x86, tuttavia, implementa il renaming dei registri, la rimozione della dipendenza logica dei dati (compresa la spedizione anti-





patata di operandi e risultati) e il bypassing dei dati per risolvere effettivamente la dipendenza dei dati e consentire l'esecuzione parallela di istruzioni che contengono queste dipendenze.

Branch control. Le istruzioni di salto si presentano in media ogni 4/6 istruzioni nei programmi in codice x86. Gli stadi della pipeline possono arrestarsi mentre si aspetta che la CPU processi il nuovo flusso di istruzioni. Il 6x86 minimizza il degrado delle prestazioni e il tempo di latenza delle istruzioni di salto attraverso l'uso della predizione di salto e dell'esecuzione speculativa.

Il 6x86 utilizza un Branch Target Buffer (BTB) a 256 posizioni, quattro vie di tipo associativo per memorizzare gli indirizzi di salto e le informazioni di predizione salto. Utilizza inoltre uno stack ad otto posizioni per memorizzare l'indirizzo di target delle istruzioni RET. La decisione di prelevare o meno l'indirizzo di target è basata su un algoritmo di branch prediction a quattro stadi che raggiunge un'accuratezza del 90%.

Floating Point Unit

L'unità in virgola mobile raggiunge alte prestazioni eseguendo le istruzioni floating

point in parallelo con le istruzioni intere attraverso un interfacciamento a 64 bit. È compatibile con il set di istruzioni x87 e aderisce allo standard IEEE-754. La FPU incorpora due code a quattro posizioni (per le istruzioni e per la memorizzazione) per facilitare l'esecuzione parallela delle istruzioni. Le informazioni sono scambiate con l'FPU attraverso otto registri accessibili sotto forma di stack, un registro di controllo e un registro di stato.

Cache Unit

Il 6x86 incorpora due cache: una unificata da 16 Kbyte (dual port) per dati e istruzioni e una Instruction Line Cache da 256 byte. La cache unificata può memorizzare istruzioni e dati in ogni proporzione e offre un migliore sfruttamento rispetto a due separate aree di memoria di egual misura per dati e istruzioni. Un incremento delle prestazioni di interfacciamento della cache con l'unità intera è dato dalla presenza di una Instruction Line Cache, fully associative, ad alta velocità.

Memory Management Unit

L'unità di Memory Management (MMU)

traduce l'indirizzo lineare fornito dall'unità intera nell'indirizzo fisico utilizzato dalla cache unificata e dal bus interface. Le procedure di Memory Management sono compatibili x86 e aderiscono al meccanismo di paginazione standard.

Bus Interface Unit

L'unità di interfacciamento bus (BIU) mette a disposizione i segnali e le temporizzazioni richieste dalla circuiteria esterna. Il bus esterno a 64 bit supporta due differenti modi di sequenza di indirizzamento per il ciclo di burst. Il modo «one-plus-four» è compatibile con le specifiche P54C (Pentium).

Operando in modalità «linear burst» la CPU minimizza l'attività di bus e offre performance migliori. Quest'ultima modalità è supportata di molti chip set a 64 bit.

Il System Management Mode (SMM) offre un interrupt che può essere utilizzato per il power management o per l'emulazione trasparente in software di periferiche di I/O.

Inoltre, il 6x86 supporta un'interfaccia hardware che permette alla CPU di entrare in modalità basso consumo (suspend mode).