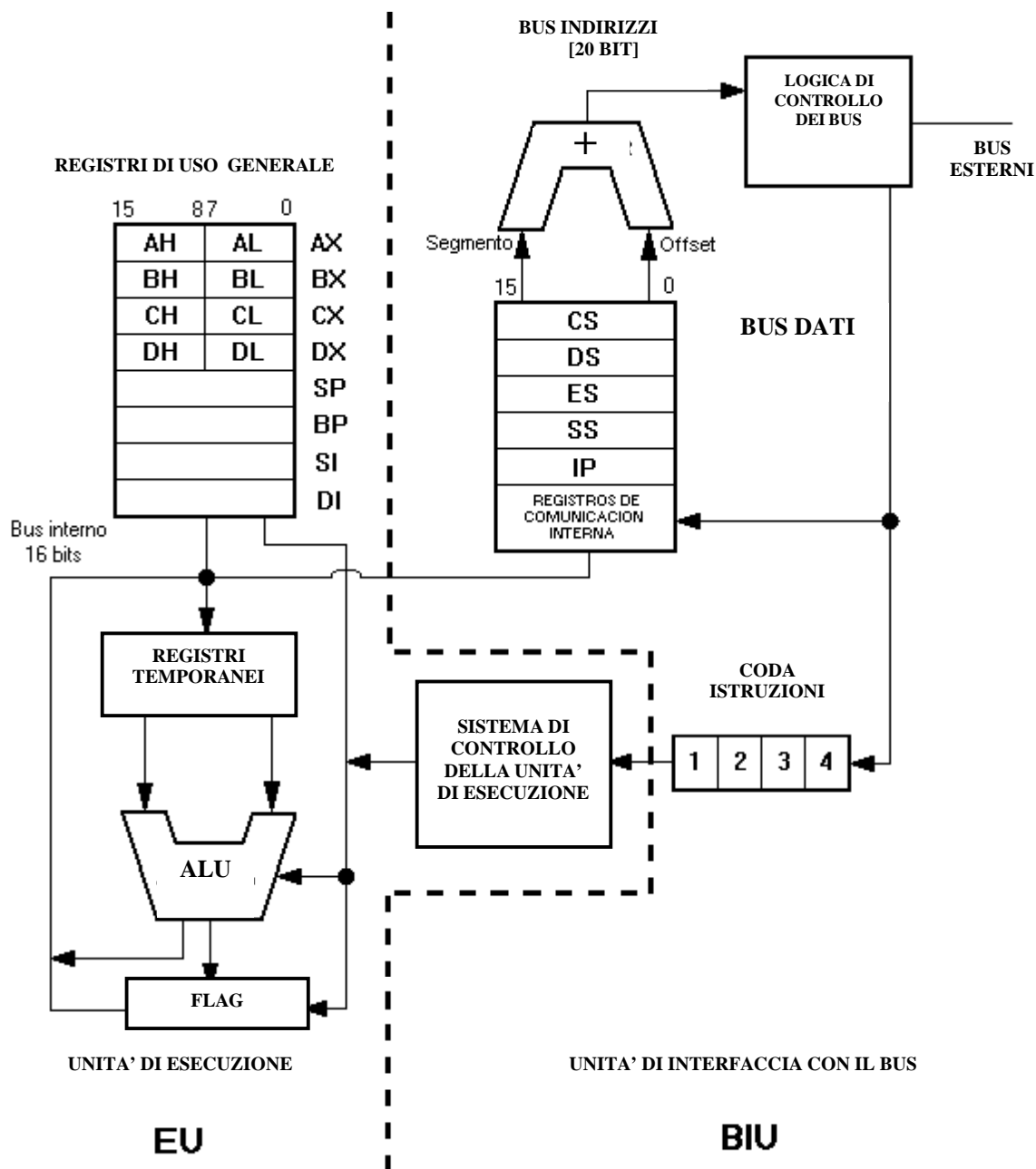


Organizzazione interna del 8086



Dal diagramma logico mostrato in figura vediamo che il microprocessore è organizzato in due zone funzionali: la Bus Interface Unit (BIU) o unità di interfaccia con il bus e la Execution Unit (EU) o unità di esecuzione. Questi due sistemi comunicano attraverso un bus a 16 bit.

Non si tratta soltanto di una distinzione logica poiché i due blocchi sono effettivamente in grado di operare parallelamente e indipendentemente l'uno dall'altro.

La BIU gestisce le comunicazioni con l'esterno e quindi l'accesso alla memoria e alle porte di ingresso uscita.

Essa si occupa di generare gli indirizzi fisici delle locazioni di memoria mediante un sommatore che effettua la somma fra il contenuto del registro di offset e il contenuto del registro di segmento shiftato a sinistra di 4 posizioni (vedi [segmentazione della memoria](#)).

Genera poi i segnali di controllo necessari per gestire le operazioni di fetch delle istruzioni e degli operandi, lettura e scrittura.

Esiste poi una zona di memorizzazione detta coda delle istruzioni in grado di memorizzare un certo numero di istruzioni: sei per il microprocessore 8086 e 4 per il microprocessore gemello 8088. Questa coda è organizzata con una struttura FIFO (First In First Out), quindi la prima istruzione ad entrare nella coda sarà la prima ad uscire.

Essa serve per effettuare il cosiddetto prefetching delle istruzioni. Ricordiamo che nel modello classico concettuale di un microprocessore, al fetching cioè al prelievo delle istruzioni dalla memoria segue la fase di esecuzione. Nel caso del prefetching l'idea è che, se l'unità di esecuzione sta eseguendo un'operazione che non richiede accessi in memoria, perché, ad esempio i dati sono contenuti tutti in registri interni del microprocessore, i bus esterni rimangono liberi e la BIU sarebbe inutilmente inattiva. Nel caso del 8086, allora, questi tempi morti sono recuperati perché, mentre la EU esegue la sua istruzione, la BIU comincia a prelevare le istruzioni seguenti presenti nella zona di memoria che contiene il codice e le incamera nella coda. Così quando la EU ha terminato di eseguire l'istruzione corrente, essa troverà già disponibile la successiva istruzione da eseguire all'interno della coda. Si eliminano così i tempi morti e si ottimizza lo sfruttamento del bus.

Questo metodo perde il suo vantaggio, naturalmente, quando si debba eseguire un'istruzione di salto ad una nuova zona di memoria per cui le istruzioni già memorizzate

nella coda non vanno eseguite. In tal caso il prefetching diventa inutile e la coda va svuotata.

La EU presenta una unità di controllo che si occupa di decodificare il codice operativo dell'istruzione da eseguirsi e generare gli opportuni segnali di controllo ai vari dispositivi presenti nel microprocessore affinché venga eseguita l'istruzione.

Vi sono poi i registri di uso generale coinvolti nelle varie operazioni, i registri SP e BP per la gestione dello stack, i registri SI e DI per la gestione delle operazioni di trasferimento dati fra zone di memoria, la unità logico-aritmetica e il registro dei flag.