

1. Risposte alle domande per l'orale

Quali sono i registri fondamentali della CPU?

I registri fondamentali, presenti in tutte le CPU (pur con nomi differenti a seconda dell'azienda produttrice di microprocessori), sono

- **AR** (Address Register): memorizza gli indirizzi per gli accessi in memoria
- **DR** (Data Register): memorizza i dati provenienti dalla memoria diretti alla CPU e dalla CPU diretti alla memoria.
- **IR** (Instruction Register): memorizza il codice operativo (opcode) dell'istruzione da eseguire.
- **PC** (Program Counter): memorizza l'indirizzo della prossima istruzione da eseguire.
- **SR** (Status Register): memorizza, tramite una serie di flag, lo stato del processore, successivo all'esecuzione dell'ultima operazione
- **SP** (Stack Pointer): memorizza l'indirizzo top dello stack. Lo stack è un'area della memoria in cui i dati sono letti/scritti in modalità Last-In-First-Out (LIFO).
- **R0, ..., Rn**: registri di lavoro che memorizzano i risultati temporanei in ingresso e in uscita dall'ALU.

2. Risposte alle domande per l'orale

Descrivi nel dettaglio cosa succede nella fase ID (Instruction Decode) del ciclo macchina.

L'opcode dell'istruzione è decodificato e in base a esso si caricano nei registri di lavoro (R0, ..., Rn) gli operandi necessari all'istruzione e si attivano le microcircuiterie necessarie a svolgere l'operazione richiesta.

3. Risposte alle domande per l'orale

Quali sono i principali problemi di gestione delle pipeline?

Questa tecnica funziona molto bene se non vi sono legami troppo stretti tra due istruzioni; in particolare se a un'istruzione serve il risultato di un'istruzione precedente, essa non potrà entrare nella fase EX fino a quando l'istruzione precedente non è giunta alla fase WB.

Un altro problema che si presenta è quello conseguente ai cosiddetti *salti di esecuzione*: blocchi di istruzioni non sono eseguite se non sono verificate determinate condizioni; in questo caso il microprocessore passa a eseguire istruzioni memorizzate "distanti" dalle precedenti con la necessità di svuotare la pipeline prima di eseguire il salto, con conseguente rallentamento dell'esecuzione.

4. Risposte alle domande per l'orale

Il processo che traduce il codice assembly nel corrispondente codice macchina eseguibile direttamente dal microprocessore, si compone di due passaggi. Descrivili.

Il processo che traduce il codice assembly nel corrispondente codice macchina eseguibile direttamente dal microprocessore si compone di due passaggi:

- **ASSEMBLER**: è un processo di traduzione che non richiede alcuna intelligenza; esiste il processo inverso, detto disassemblaggio o disassembly;
- **LINKER**: serve a collegare moduli e librerie di cui si compone il programma e a distribuire il codice oggetto nello spazio di indirizzi di memoria centrale assegnato al programma.

5. Risposte alle domande per l'orale

Quali sono e cosa indicano i 6 flag di stato del registro di stato?

I flag di stato sono:

- *C* (Carry): è settato a 1 in presenza di riporto (carry) o prestito (borrow) a seguito di operazioni aritmetiche nell'ALU, altrimenti è resettato a 0;
- *P* (Parity): è il bit di parità; in parità pari è settato a 1 per avere un numero pari di bit a 1 nel dato, viceversa in parità dispari; serve per la rilevazione degli errori di trasmissione;
- *A* (Auxiliary): è usato per carry e borrow da un *nibble* (pari a un semibyte, cioè 4 bit) a quello immediatamente successivo;
- *Z* (Zero): è settato a 1 quando il risultato di un'operazione aritmetica è 0, altrimenti è resettato 0;
- *S* (Sign): è settato a 1 quando il risultato di un'operazione aritmetica è negativo (in complemento a 2 il bit segno vale 1 per i numeri negativi), altrimenti è resettato 0;
- *O* (Overflow): settato a 1 segnala un "traboccamento", cioè segnala se il risultato di un'operazione aritmetica tra interi produce un risultato che va a occupare il bit dedicato al segno del numero stesso.

6. Risposte alle domande per l'orale

Descrivi il funzionamento dell'istruzione LOOP.

L'istruzione LOOP serve per l'esecuzione e il controllo dei cicli. Ve ne sono di tre tipi:

- LOOP: esegue il ciclo se CX è diverso da 0;
- LOOPE: preceduta dalla CMP, esegue il ciclo se CX è diverso da 0 e il flag Z = 0;
- LOOONE: preceduta dalla CMP, esegue il ciclo se CX è diverso da 0 e il flag Z = 1.

7. Risposte alle domande per l'orale

Come si dichiara una subroutine?

Le subroutine (procedure) si dichiarano nel formato *Nome_sub PROC NEAR*, seguito dalle istruzioni che devono terminare con l'istruzione *RET* (per il ritorno alla subroutine o al main chiamante), seguita dalla parola chiave *endp* (end procedure); le PROC NEAR si scrivono in fondo al code segment. Se si lavora a moduli (su più file .asm) e la subroutine chiamata non è nello stesso file del chiamante, occorre dichiarare la subroutine come: *Nome_sub PROC FAR*.

8. Risposte alle domande per l'orale

I metodi indirizzamento si suddividono in tre tipi. Quali?

I metodi di indirizzamento si suddividono in tre tipi, in base al modo in cui l'istruzione tratta il dato da elaborare:

- *immediato*: nell'istruzione c'è il dato;
- *diretto*: nell'istruzione c'è l'indirizzo di memoria o il registro in cui si trova il dato;
- *indiretto*: nell'istruzione c'è l'indirizzo di memoria o il registro dove si trova l'indirizzo di memoria in cui c'è il dato.

9. Risposte alle domande per l'orale

Che differenza c'è tra indirizzamento immediato e indirizzamento diretto?

L'indirizzamento immediato coinvolge un solo registro e non consente l'inversione degli operandi. Invece l'indirizzamento diretto coinvolge uno o due registri o una locazione di memoria.