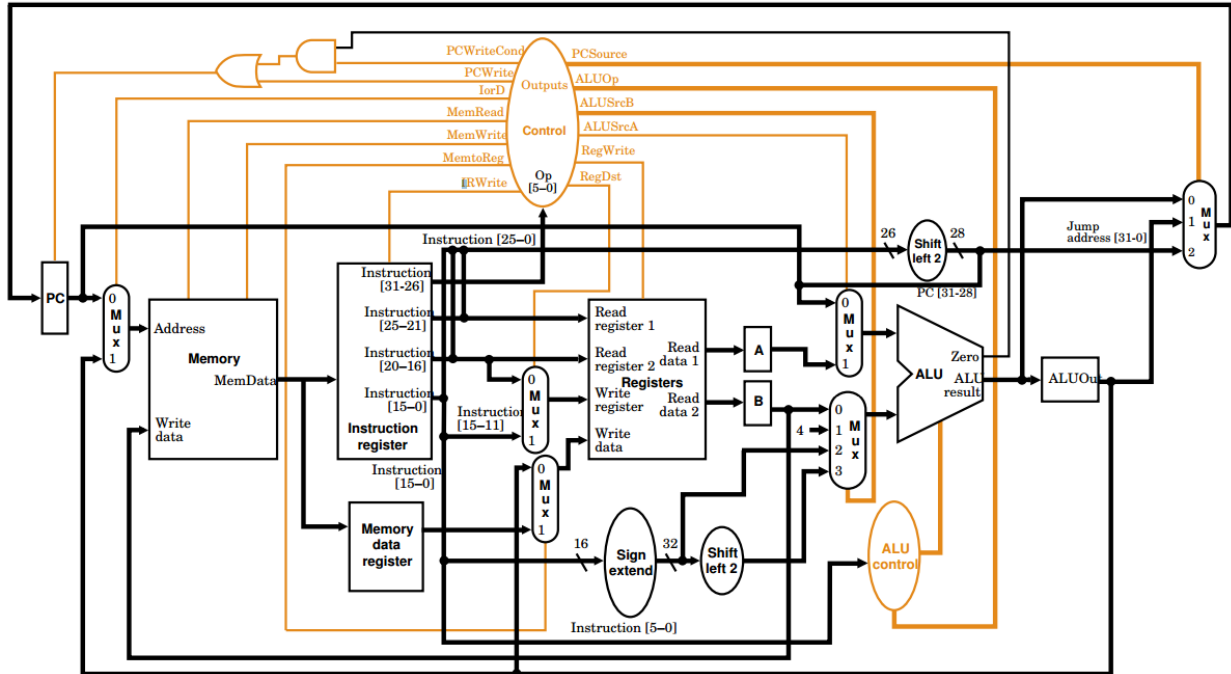


1. [7] Si consideri la CPU seguente, che esegue il codice a lato. Determinare il contenuto del registro MDR (in formato esadecimale) ad ogni ciclo di clock necessario per eseguire interamente il codice riportato;

```

0xCCC: addi $6, $0, 0xA0
        sw $6, 80($6)
        add $6, $6, $6
        lw $10, 64($6)
    
```



2. [5] Un processore caratterizzato da uno spazio di memoria indirizzabile di 64 GByte e un bus dati di 64 bit viene dotato di una memoria cache a quattro vie, di capacità totale 16 MByte e con linee di 16 parole. a) Calcolare esattamente quanti bit di memoria contiene l'intera cache (considerando anche quelli per memorizzare il bit di validità e il campo tag); b) calcolare il valore (in esadecimale) di: *byte offset*, *word offset*, *index* e *tag* corrispondenti all'indirizzo di memoria principale: **0xABCDEF**.
3. [6] Mostrare come vengono effettuate l'allocazione statica e l'allocazione dinamica in Assembly MIPS, scrivendo, in Assembly MIPS nativo:
- il frammento di codice che alloca staticamente un array di 100 parole e pone l'indirizzo base nel registro \$7, inizializzando ciascuna parola con il proprio numero di indice;
  - il frammento di codice che alloca dinamicamente un array di N parole, dove il numero N viene richiesto all'utente da tastiera e l'indirizzo del registro base viene posto nel registro \$7.
4. [4] Si disegni la struttura circuitale (per blocchi funzionali) di una memoria con controllo d'errore, con parole da 16 bit e codice rivelatore d'errore di 2 bit.
5. [3] Elencare e descrivere i criteri più comuni utilizzati nelle memorie cache a più vie per la scelta del banco in cui scrivere un nuovo blocco.
6. [5] Disegnare la struttura circuitale interna (a livello di blocchi funzionali fondamentali) dell'unità di propagazione (*Forwarding unit*) della CPU MIPS32 pipeline, e descriverne il funzionamento.

