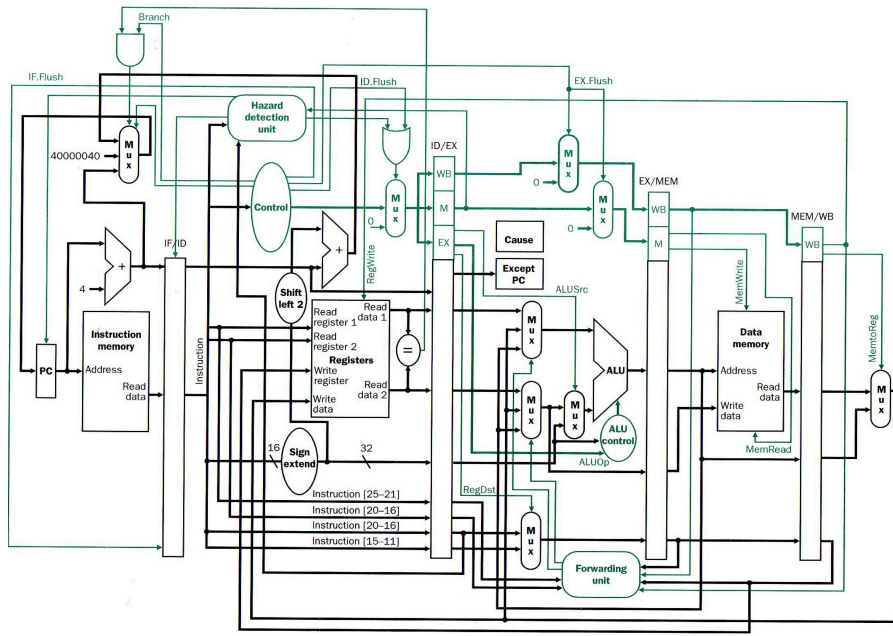




1. [7] Si consideri la CPU seguente mentre esegue il codice a lato.
- Identificare gli eventuali casi di criticità nel codice e descrivere come vengono gestiti dalla CPU;
 - determinare il contenuto del registro **ID/EX** (in uscita) dopo **4 cicli di clock** dall'inizio della prima istruzione;
 - determinare quanti cicli di clock in totale sono necessari per completare tutte le istruzioni.

```
0xAAC: sw $4, 12($3)
        lw $4, 16($3)
        addi $3, $3, 1
        sw $3, 0($4)
```



2. [5] Un processore caratterizzato da una capacità di indirizzamento di 1 GB e da un bus dati di 16 bit viene dotato di una memoria cache associativa a mappatura diretta, di capacità totale $C = 2$ MByte e con linee di 32 parole. a) Dimensionare la cache determinando le dimensioni di tutti i campi; b) disegnare lo schema circuitale dettagliato di tale memoria; c) calcolare la posizione nella cache in cui viene memorizzato il byte di memoria situato all'indirizzo: **0x0...080A0**, determinando i valori in esadecimale di: *byte offset*, *word offset*, *index* e *tag*.
3. [5] Spiegare come funziona la memoria virtuale, descrivendo come avviene, a livello circuitale, l'accesso alla memoria attraverso l'uso della *page table* e del *page table register*.
4. [5] Si incrementano le prestazioni di calcolo di un elaboratore mediante l'introduzione di una FPU, che esegue calcoli in virgola mobile in 2 nsec; senza di essa la CPU esegue gli stessi calcoli in 42 nsec. a) Quanto durerà ora l'esecuzione di un programma che esegue calcoli in virgola mobile per lo 87,5% del tempo, se originariamente impiegava 30 secondi. b) Con che percentuale di calcoli in virgola mobile si otterrebbe una durata del programma di 10 secondi?
5. [4] Calcolare il tempo massimo di refresh ammesso da un chip di memoria dinamica di 16M celle da 1 bit, se il tempo di scarica della cella è di 102,4 millisecondi.

6. [6] Rappresentare il contenuto, byte per byte in formato esadecimale, e gli indirizzi corrispondenti, della zona di memoria che viene modificata a seguito dell'esecuzione del frammento di codice a lato (si ricorda che il codice ASCII di "A"=65).

Determinare il valore contenuto nel registro **\$v0** al ritorno dalla chiamata **syscall**.

```
.data 0x490
.byte -40
.word -15, 0x20
.space 12
.asciiz "ABC"
.text
main: addi $a0, $zero, 40
      addi $v0, $zero, 9
      syscall
```