



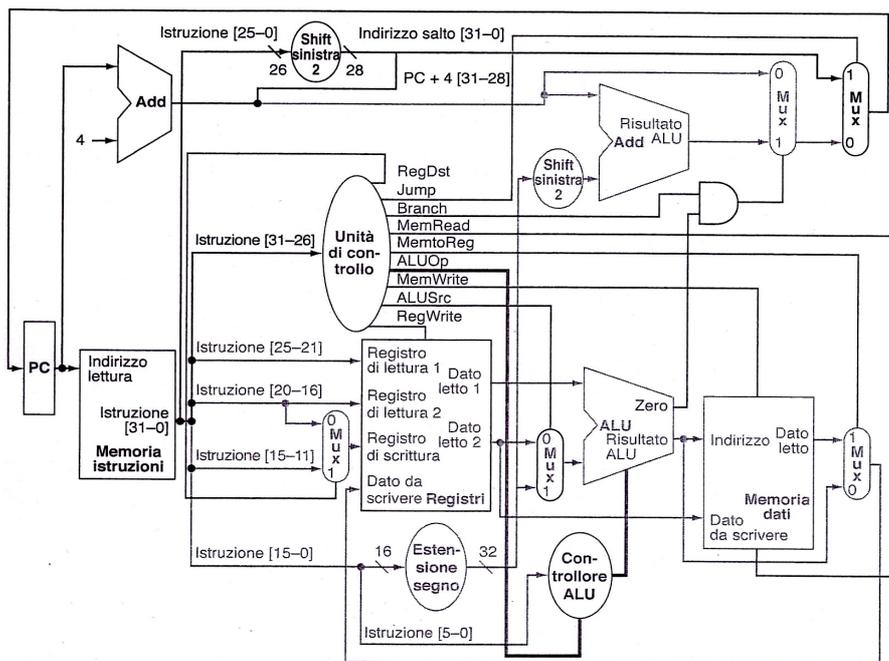
- [3] A che numero corrisponde la parola **0xC3CE000** se codificata secondo lo standard IEEE-754, singola precisione?
- [4] Mediante passaggi algebrici, trasformare la funzione logica $f(a,b,c,d) = a + b + \bar{d}$ nella sua seconda forma canonica.
- [4] Si progetti un circuito caratterizzato da 4 linee in ingresso ($i_3 i_2 i_1 i_0$) che rappresentano un numero intero con segno I (utilizzando la notazione in complemento a 2), e da un'uscita U che vale '1' se e solo se I , in valore assoluto, è minore di 3 o divisibile per 3. a) Determinare la tabella di verità delle uscite; b) esprimerle nella forma canonica più adatta; c) semplificarle mediante mappe di Karnaugh; d) semplificarle ulteriormente, se possibile, mediante passaggi algebrici; e) disegnarne il circuito.
- [4] Si disegni la struttura circuitale di una ALU a 4 bit in grado di eseguire le operazioni: AND, OR, +, -, LESS (come nella ALU MIPS).
- [5] Si traduca il seguente frammento di codice Assembly MIPS in linguaggio macchina, in formato esadecimale. Si calcolino inoltre i due indirizzi effettivi di salto di ciascuna delle due istruzioni.

<pre>0x6789012C: beq \$t0,\$t1,0xFFD8 j 160</pre>	<pre>(OpCodes: beq=4, j=2); (\$t0=\$8)</pre>
---	--
- [6] Si sintetizzi una FSM di Moore sincrona caratterizzata da una linea d'ingresso, osservata ogni millisecondo, ed una linea di uscita che funziona come segue: quando l'ingresso passa da '0' a '1' e rimane a '1' per 3 millisecondi o più, l'uscita cambia di valore quando l'ingresso torna a '0'. Se invece l'ingresso torna a '0' prima dei 3 millisecondi, l'uscita rimane invariata. Si assumano, allo stato iniziale, ingresso e uscita entrambi a '0'. Si determinino: STG, STT, STT codificata e le funzioni uscita e stato prossimo, semplificandole il più possibile. Si disegni infine il circuito complessivo della macchina.

- [6] Esprimere in esadecimale e con il corretto numero di bit, i valori in ingresso e in uscita da:
 - ogni ALU,
 - dai MUX relativi ai salti,
 - dal Register File,
 supponendo che la CPU stia in questo momento eseguendo la seguente istruzione:

0x8C: addi \$5, \$0, -20
(OpCode addi: 8)

Rappresentare inoltre l'istruzione in linguaggio macchina, in formato esadecimale.



Nota: si prega di svolgere questo esercizio sul proprio foglio, disegnando ALU, MUX e Register File, e scrivendo i valori richiesti in corrispondenza dei rispettivi ingressi e uscite.