

Cognome e nome dello studente:

Matricola:

Anno di corso e turno:

A.A. 2004-2005 – Appello del 14 Febbraio 2006

1. [6] Si consideri una cache a mappatura diretta di capacità  $C = 32$  kByte e con linee contenenti 8 parole, utilizzata da una CPU caratterizzata da un bus indirizzi di 24 bit e bus dati di 16 bit. Dimensionare e disegnare lo schema della cache, mettendo in evidenza come le diverse parti dell'indirizzo di memoria centrale controllino il circuito. Indicare in che posizione viene scritto il byte con indirizzo in memoria centrale 0xFA8420.

2. [5] Supponendo che la CPU in figura debba eseguire soltanto le seguenti istruzioni (in formato MIPS):

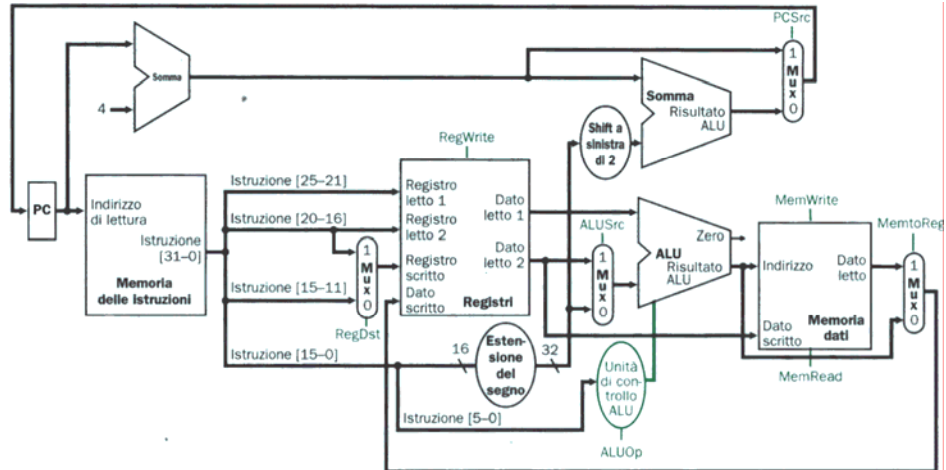
**add** \$s0, \$s1, \$s2

**lw** \$t0, 80(\$t1)

**j** label1

**beq** \$s0, \$t0 label2

progettare l'unità di controllo.



Codici operativi: add -> 8; lw -> 35; j -> 2; beq -> 4.

3. [7] Si sintetizzi una macchina a stati finiti di Moore (rappresentando STG, STT e struttura circuitale) che riceve in ingresso sequenze di bit e presenta due linee d'uscita, U1 ed U2. La linea U1 assume il valore '1' soltanto se sia stata ricevuta la sequenza '000', mentre U2 va a '1' solo quando si riceve la sequenza '1111'. Lo stato iniziale della macchina sia quello di sequenza vuota.
4. [6] Si traduca in linguaggio Assembly il seguente frammento di codice, identificando eventuali procedure-foglia. (N.B. nel prodotto, si consideri la parola meno significativa; si specifichi, nei commenti, la corrispondenza tra le variabili C e Assembly)

```

int sommaquad( unsigned int n )
{
    unsigned int s;
    if ( n == 0 )    s = 0;
    else
        s = quad(n) + sommaquad( n-1 );
    return( s );
}

int quad( unsigned int v )
{
    unsigned int vq = v*v;
    return( vq );
}
    
```

5. [3] Descrivere, mediante diagramma di flusso, un algoritmo di moltiplicazione "firmware" e disegnarne la corrispondente struttura circuitale.
6. [3] Descrivere le strutture necessarie in una CPU per la gestione di interrupts e le principali strategie di gestione della risposta allo stesso.
7. [2] Quando e perché è necessario un arbitraggio nella gestione di un bus? Descrivere le più comuni tecniche di arbitraggio utilizzate nei bus dei calcolatori. Principi di funzionamento del bus IEEE1394 firewire.
8. [4] Modificare lo schema circuitale riportato sopra in modo tale che le stesse istruzioni siano eseguibili in pipeline. Quali criticità vedete nell'esecuzione e motivare la risposta.