



1. [4] Sintetizzare la funzione logica nella prima e seconda forma canonica associata alla seguente espressione logica:  $Y = AB + \overline{AC} + \overline{BC}$ . Dimostrare che le due forme canoniche sono equivalenti trasformando la prima forma canonica nella seconda.

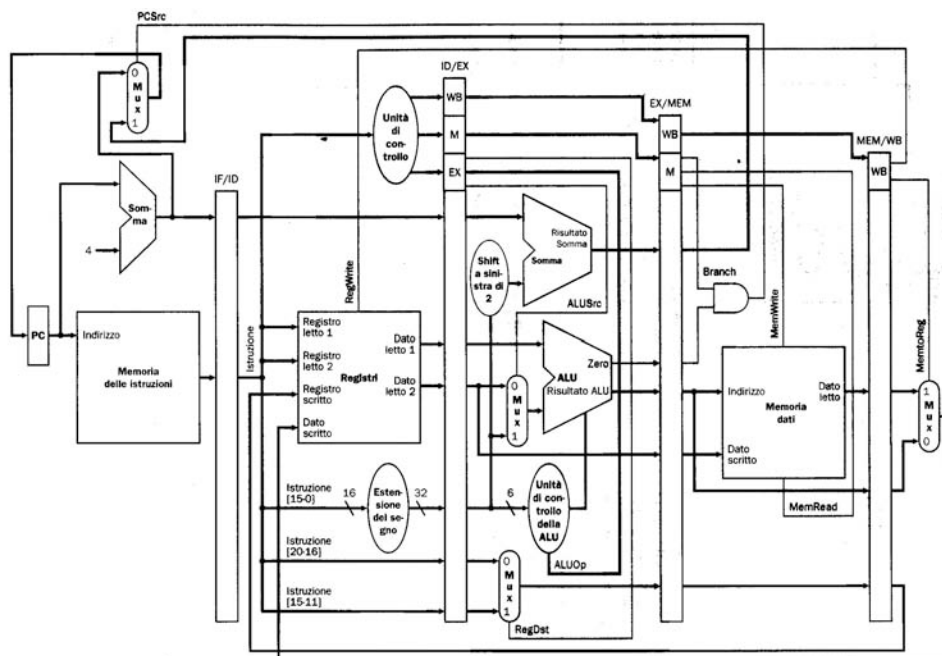
2. [6] Tradurre in linguaggio Assembly MIPS la seguente funzione, evitando di utilizzare pseudo-istruzioni:

```
int BellaFunz( int n )           // n è un intero positivo
{
    if( n < 10 )
        return( n + BellaFunz(2*n) );
    else
        if( n > 100 )
            return( n + BellaFunz( n/2 ) );
        else return( n );
}
```

3. [7] Data l'architettura in figura, scrivere il contenuto dei registri di pipeline (l'uscita dei registri) durante l'esecuzione della `lw` in questo frammento di codice:

```
0x100:
add $s4, $s1, $s2
sub $t0, $t1, $t2
lw $s0, 0($s5)
beq $s0, $s2, 100
add $s6, $s6, $s6
```

Ci sono hazard?  
Quali?



4. [7] Progettare una macchina a stati finiti di Moore che analizzi sequenze binarie da un ingresso seriale e sia caratterizzata da tre uscite, di cui:

- la prima si porta a "1" quando gli ultimi 2 bit della sequenza in ingresso siano stati "1";
- la seconda si porta a "1" quando gli ultimi 3 bit della sequenza in ingresso siano stati "1";
- la terza si porta a "1" quando gli ultimi 4 o più bit della sequenza in ingresso siano stati "1".

Si supponga che nello stato iniziale la sequenza sia vuota. Determinare STG, STT, STT codificata e la struttura circuitale completa della macchina.

5. [5] Si progetti e si disegni lo schema dettagliato (esplicitando le dimensioni di tutti i campi) di una memoria cache a 2-associativa, che serve un processore con bus dati di 8 bit e bus indirizzi di 20 bit. La capacità totale della cache sia di 16 kbyte e la dimensione del blocco di 16 parole.

6. [4] Dare la definizione di *eccezione* ed *interrupt* e descrivere i più comuni metodi di gestione della risposta a tali eventi.

7. [3] Cosa è un *branch prediction buffer*? Come funziona?