

# Architettura dei Calcolatori

## Prova scritta – 10 settembre 2019 – 1h30

PARTE 1 – RISPOSTA SINGOLA - Ogni domanda ha una sola risposta VERA.

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
- Una risposta errata fa perdere il punteggio negativo riportato a fianco della domanda
- Una risposta lasciata in bianco viene valutata 0

1. (2, -.5) Si consideri un display a colori con una risoluzione di 1024x1024 pixel, e che rappresenta ogni pixel con 8 bit per ciascun colore primario (rosso, verde, blu). Quale dimensione minima in MBytes deve avere un frame buffer per poter memorizzare una immagine a pieno schermo?

a) 0,375

b) 1

c) 3

d) Nessuna delle precedenti

$$\begin{aligned} & 1024 * 1024 \text{ pixel} * 8 \frac{\text{bit}}{\text{colore}} * 3 \frac{\text{colori}}{\text{pixel}} = \\ & = (1024 * 1024 * 8 * 3) \text{ bits} * \frac{1 \text{ Byte}}{8 \text{ bits}} * \frac{1 \text{ KBytes}}{1024 \text{ Byte}} * \frac{1 \text{ Mbytes}}{1024 \text{ Kbytes}} = \\ & = \frac{(1024 * 1024 * 8) * 3}{(1024 * 1024 * 8)} \text{ MBytes} = \mathbf{3 \text{ MBytes}} \end{aligned}$$

2. (2, -.5) I registri x5 e x6 di un processore RISC-V contengano i valori 0x8000 0000 0000 0000 e 0xD000 0000 0000 0000, rispettivamente. Quale affermazione è corretta?

a) Non c'è overflow se si esegue l'istruzione add x30, x5, x6

b) Non c'è overflow se si esegue l'istruzione sub x30, x6, x5

c) Non c'è overflow se si esegue l'istruzione sub x30, x5, x6

d) Nessuna delle precedenti

3. (2, -.5) Si assuma che gli stadi individuali del datapath di una CPU RISC-V abbiano le seguenti latenze: IF=250ps; ID=350ps; EX=150ps; MEM=300ps; WB=200ps. Qual è il tempo di un ciclo di clock per una CPU pipelined e per una non pipelined?

a) 350ps per la CPU pipelined e 1250ps per la CPU non pipelined

b) 150ps per la CPU pipelined e 1250ps per la CPU non pipelined

c) 350ps per la CPU pipelined e per la CPU non pipelined

d) Nessuna delle precedenti

4. (2, -.5) Il rapporto tipico della latenza di lettura di un dato da una cache di primo livello rispetto alla lettura da RAM è

a) 1 a 1 (ovvero, la latenza è la stessa)

b) 1 a 10 (ovvero, la RAM è dieci volte più lenta della cache L1)

c) 1 a 100

d) 1 a 10000

## PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE -

Ogni domanda può avere da una a quattro risposte CORRETTE.

- Ogni risposta esatta viene calcolata: +1
  - Ogni risposta errata viene calcolata: -0.5
  - Una risposta lasciata in bianco viene calcolata: 0
5. Si considerino tre processori P1, P2 e P3, caratterizzati dallo stesso *instruction set*, ma con frequenze e CPI diversi, e precisamente:  
P1: freq = 3,0GHz, CPI = 1,5.  
P2: freq = 2,5GHz, CPI = 1,0.  
P3: freq = 4,0GHz, CPI = 2,2.

Quali affermazioni sulla performance (istruzioni al secondo) sono corrette?

- a) *Il processore P1 ha la performance peggiore, con 2 miliardi di istruzioni al secondo*
- b) *Il processore P2 ha la performance migliore, con 2,5 miliardi di istruzioni al secondo*
- c) *Il processore P3 ha la performance peggiore, con 1,8 miliardi di istruzioni al secondo*
- d) *Nessuna delle precedenti*

6. Il numero ABCDEF in esadecimale:

- a) Vale 101112131415 in decimale
- b) *Vale 101010111100110111101*
- c) *Vale 52746757 in ottale*
- d) Nessuna delle precedenti

$$P1 = \left(3,0 * 10^9 \frac{\text{cicli}}{\text{sec}}\right) * \left(\frac{1 \text{ istruzioni}}{1,5 \text{ cicli}}\right) = 2,0 * 10^9 \frac{\text{istruzioni}}{\text{sec}}$$
$$P2 = \left(2,5 * 10^9 \frac{\text{cicli}}{\text{sec}}\right) * \left(\frac{1 \text{ istruzioni}}{1,0 \text{ cicli}}\right) = 2,5 * 10^9 \frac{\text{istruzioni}}{\text{sec}}$$
$$P3 = \left(4,0 * 10^9 \frac{\text{cicli}}{\text{sec}}\right) * \left(\frac{1 \text{ istruzioni}}{2,2 \text{ cicli}}\right) = 1,8 * 10^9 \frac{\text{istruzioni}}{\text{sec}}$$

7. Un gruppo di studenti discute dell'efficienza di una pipeline a 5 stadi, quando uno di loro fa presente che non tutte le istruzioni usano tutti gli stadi della pipeline. Dopo aver deciso di ignorare gli effetti degli hazard, gli studenti traggono le seguenti conclusioni. Quali sono corrette?
- a) *Consentire alle istruzioni di tipo **ALU** e **branch** di usare meno stadi dei cinque richiesti dalla **load** migliorerebbe la performance della pipeline in tutte le circostanze.*
  - b) *Cercare di ridurre il numero di cicli per alcune istruzioni non aiuta, dato che il throughput è determinato dal periodo di clock; il numero di stadi di pipeline per istruzione impatta la latenza, non il throughput.*
  - c) *Non si può far sì che le istruzioni ALU impieghino meno cicli per via del writeback dei risultati. I branch però possono impiegare meno cicli, quindi c'è margine di miglioramento.*
  - d) *Piuttosto che cercare di far eseguire le istruzioni in meno cicli, sarebbe opportuno cercare di rendere la pipeline più lunga. In questo modo le istruzioni richiederebbero più cicli per eseguire, ma i cicli sarebbero più corti. Questo migliorerebbe la performance.*

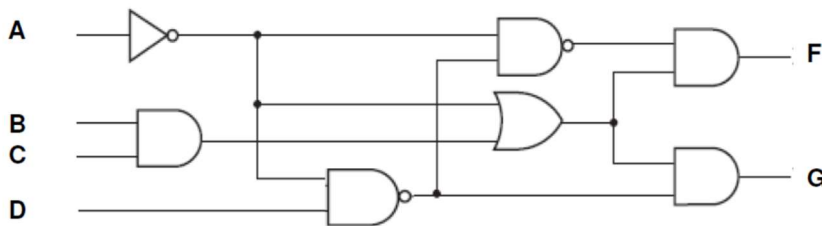
8. DOMANDA SUL BLOCCO MEMORY HIERARCHY

- a)
- b)
- c)

### PARTE 3 – DOMANDE APERTE

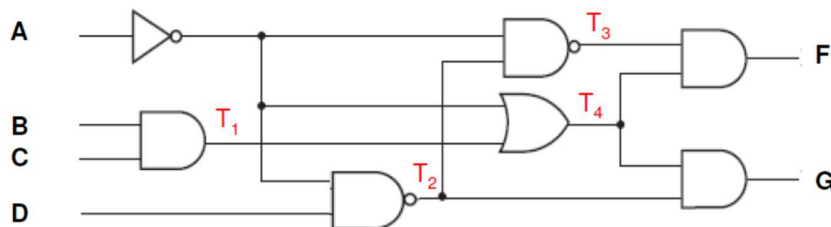
- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
- Una risposta errata può eventualmente causare una penalità che dipende dalla gravità dell'errore
- Una risposta lasciata in bianco viene calcolata: 0
- SI RICORDA CHE L'UNICO FOGLIO DA CONSEGNARE E' IN CALCE AL COMPITO. QUESTO FOGLIO, PUO' SERVIRE ESCLUSIVAMENTE COME "BRUTTA COPIA". EVENTUALI RISPOSTE SCRITTE IN QUESTO FOGLIO NON VERRANNO PRESE IN CONSIDERAZIONE

9. (6 pt) Analizzare il circuito seguente. Determinare le espressioni booleane per le uscite F e G in funzione degli ingressi A, B, C, D.



Se possibile, si minimizzino le espressioni ricavate tramite metodologia di Karnaugh

#### SOLUZIONE



**PRIMA PARTE (4 PUNTI)** - Determiniamo le corrispondenti funzioni:

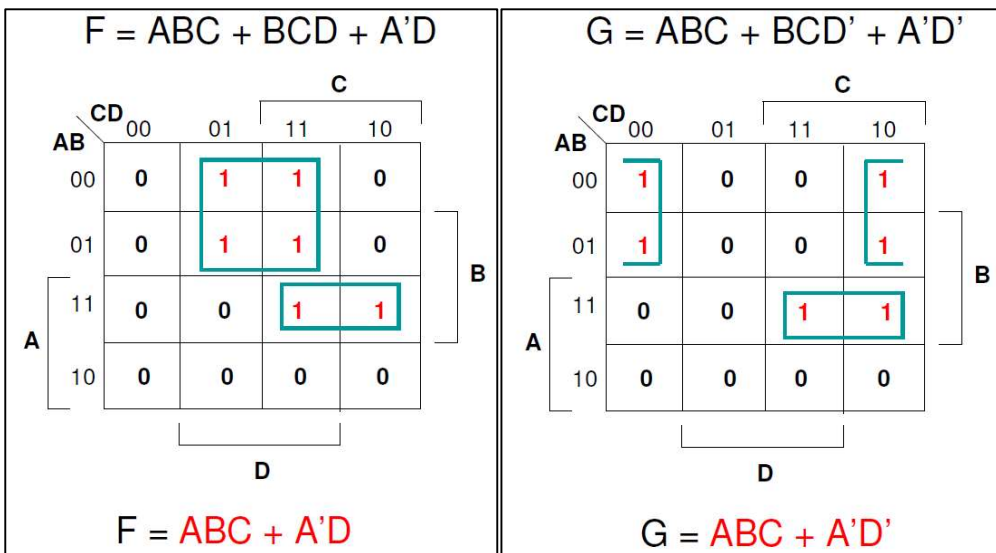
- $T_1 = BC$
- $T_2 = (A'D)'$
- $T_3 = (A'T_2)'$
- $T_4 = (A'+T_1)$

Determiniamo le funzioni delle uscite in forma di somma di prodotti, sostituendo le funzioni intermedie con le loro espressioni:

$$F = T_3T_4 = (A'T_2)'(A'+T_1) = [A'(A'D)']'(A'+BC) = [A+(A'D)](A'+BC) = (A+D)(A'+BC) = ABC + BCD + A'D$$

$$G = T_4T_2 = (A'+T_1)[(A'D)'] = (A'+BC)[(A'D)'] = (A'+BC)(A+D') = A'D' + BCD' + ABC$$

**SECONDA PARTE (2 PUNTI)** – Disegniamo le mappe di Karnaugh



10. (6 pt) Si assuma che  $x11$  contenga 11 e  $x12$  contenga 22. Supponendo di eseguire il codice sottostante su una versione della pipeline a cinque stadi RISC-V che non gestisce i data hazards (il programmatore deve gestirli esplicitamente inserendo delle NOP), si dica che valore hanno alla fine i registri  $x13$ ,  $x14$  e  $x15$ . Se questi valori sono diversi da quelli attesi, si mostri come occorre modificare il codice per ottenere i risultati corretti. Si assuma che il *register file* sia scritto durante la prima metà di un ciclo e letto durante la seconda metà (quindi ciò che viene scritto nello stadio WB a un dato ciclo è visibile nello stesso ciclo nello stadio ID)

```

addi x11, x12, 5
add x13, x11, x12
addi x14, x11, 15
add x15, x11, x11

```

addi x11, x12, 5		x11=11 x12=22			x11=27		
add x13, x11, x12			x11=11 x12=22			x13=33	
addi x14, x11, 15				x11=11		x14=26	
add x15, x11, x11					x11=27		x15=54

IF    ID    EXE    MEM    WB

addi x11, x12, 5		x11=11 x12=22					
NOP							
NOP							
add x13, x11, x12				x11=27 x12=22		x13=49	
addi x14, x11, 15					x11=27		x14=42
add x15, x11, x11						x11=27	x15=54

▲ PRIMA PARTE (4 PUNTI)

◀ SECONDA PARTE (2 PUNTI)

11. (5 pt) Quale espressione C corrisponde alle seguenti istruzioni RISC-V? Si assuma che le variabili  $f, g, h, i, j$  siano assegnate ai registri  $x5, x6, x7, x28, x29$ , rispettivamente, che  $A$  e  $B$  siano array di **double** (8 bytes), e che il loro indirizzo base sia nei registri  $x10$  e  $x11$ .

```

slli x30, x5, 3      // x30 = f*8
add x30, x10, x30    // x30 = &A[f]
slli x31, x6, 3      // x31 = g*8
add x31, x11, x31    // x31 = &B[g]
ld x5, 0(x30)        // f = A[f]
addi x12, x30, 8     // x12 = &A[f] + 8 = &A[f+1]
ld x30, 0(x12)       // x30 = A[f+1]
add x30, x30, x5     // x30 = A[f+1]+f
sd x30, 0(x31)       // B[g] = A[f+1]+f

```

$B[g] = A[f+1]+f$