

**SECONDA PROVA INTERMEDIA DEL MODULO DI
CALCOLATORI ELETTRONICI
CORSI DI LAUREA IN
INGEGNERIA ELETTRICA, ELETTRONICA E INFORMATICA
INGEGNERIA BIOMEDICA
IMMATRICOLATI A.A. 2014/15 e precedenti
20 maggio 2016**

NOME:

COGNOME:

MATRICOLA:

ESERCIZIO 1 (12 punti)

Si consideri un'immagine a livelli di grigio che viene memorizzata dal MIPS in un'area di memoria di tipo vettoriale a partire dall'indirizzo memorizzato nel registro \$4: ciascuna parola, a partire da questo indirizzo, contiene un determinato livello di grigio o pixel, compreso tra 0 e 255. L'immagine è formata in tutto da N pixel, con N memorizzato nel registro \$5. Scrivere una funzione Assembly MIPS *binarizza*, che restituisca in un'analogia area di memoria di tipo vettoriale ma a partire dall'indirizzo memorizzato in \$7, un'immagine "binarizzata", ovvero un'immagine che contenga, per ogni parola (pixel) corrispondente a quella dell'immagine data in \$4, il valore massimo (255, corrispondente al bianco) se il livello di grigio iniziale è superiore ad un valore di soglia memorizzato nel registro \$6, altrimenti contenga il minimo (0, corrispondente al nero).

Esempio di binarizzazione di immagine:



Immagine originale (livelli di grigio)
(indirizzo iniziale in \$4)



Immagine binarizzata (bianco e nero)
(indirizzo iniziale in \$7)

ESERCIZIO 2 (7 punti)

Si consideri la seguente rappresentazione di numeri nel formato binario: campo complessivo di 16 bit; virgola mobile, con mantissa frazionaria in segno e valore di tipo 1.M ed esponente a 6 bit in eccesso 32.

1. (2 punti). Esprimere il minimo ed il massimo valore rappresentabili in valore assoluto, escluso lo zero.
2. (3 punti) Rappresentare il valore 215.525.
3. (2 punti) Indicare a quanto ammonta in percentuale la perdita di precisione eventuale nella rappresentazione del valore dato rispetto al valore atteso.

ESERCIZIO 3 (6 punti)

Si progetti una ALU con operandi a N bit utilizzando un parallel adder e delle opportune reti logiche, descrivendo il relativo schema e realizzazione circuitale con le caratteristiche indicate dalla seguente tabella di verità (l'apice indica il complemento a 1 bit per bit). Si consideri in tutti e quattro i casi il c_{in} forzato a 1:

s1	s0	F
0	0	A'
0	1	A-B
1	0	B-A
1	1	B'

ESERCIZIO 4 (8 punti)

L'ampiezza della linea dati del bus di un calcolatore è pari a 32 bit. La frequenza del clock della CPU è di 2 GHz.

- 1) (3 punti) Ipotizzando che il bus sincrono abbia la stessa frequenza di clock della CPU e la durata di una trasmissione sul bus sia pari a 2 cicli di clock, e che il tempo di ciclo della memoria sia pari a 20 cicli di clock, illustrare chiaramente il protocollo di lettura su bus sincrono utilizzando l'opportuno grafico, indicando il tempo complessivo di trasferimento di una parola di 32 bit.
- 2) (3 punti) Si considerino le seguenti istruzioni Assembly MIPS assumendo che gli indirizzi di memoria calcolati coincidano con il sistema memory-mapped delle periferiche. Se le istruzioni che non richiedono trasferimenti lungo il bus dati/indirizzi impiegano il 10% del tempo che impiegano le istruzioni di trasferimento dati da/a memoria, tempo da intendersi calcolato nel punto 1, quanto tempo occorre al sistema per effettuare la stampa indicata dalle istruzioni sottostanti?

```
        addi $5, $0, 10
        move $4, $0
transfer_loop: sw $4, printer($0)    #stampa memory-mapped
              addi $4, $4, 1
              bne $4, $5, transfer_loop
```

- 3) (2 punti) Se, ad un dato istante, il numero medio di cicli necessari per l'esecuzione di un'istruzione è 4, ed il trasferimento dati avviene mediante modulo DMA con furto di ciclo a 307200 bps, di quanto viene rallentato il processore in termini di numero di istruzioni al secondo?

Soluzioni

ESERCIZIO 1

Utilizziamo i registri \$8-11 come segue:

\$8	contatore
\$9	indirizzo del pixel i dell'immagine data/indirizzo del pixel i dell'immagine binarizzata
\$10	valore del livello di grigio del pixel i dell'immagine data/valore del livello di grigio del pixel i dell'immagine binarizzata
\$11	condizione valore di soglia inferiore a quella del pixel dell'immagine data

```
binarizza:    addi $29, $29, -16 #spazio nello stack
              sw $8, 0($29)      #salvataggio del contesto
              sw $9, 4($29)
              sw $10, 8($29)
              sw $11, 12($29)
              move $8, $0        #i=0
loop:         beq $8, $5, exit    #while i<N:
              muli $9, $8, 4     # j=i*4
              addi $9, $9, $4    # j=j+ind_immagine
              lw $10, 0($9)      # prelevo immagine[j]
              slt $11, $6, $10   # binarizzo=immagine[j]>soglia
              muli $10, $11, 255 # ldgrigio=binarizzo*255
              muli $9, $8, 4     # j=i*4
              addi $9, $9, $7    # j=j+ind_imgbinarizzata
              sw $10, 0($9)      # imgbinarizzata[j]=ldgrigio
              addi $8, $8, 1     # i=i+1
              j loop
exit:         lw $8, 0($29)      #ripristino del contesto
              lw $9, 4($29)
              lw $10, 8($29)
              lw $11, 12($29)
              addi $29, $29, 16  #ripristino dello stack pointer
              jr $31            #ritorno al chiamante
```

ESERCIZIO 2

Soluzione alla domanda 1.

Il minimo valore si ha con il minimo valore dell'esponente:

minimo = 2^{-32}

Il massimo valore si ha con tutti i bit della mantissa pari ad 1, e con il massimo esponente. Poiché il bit di segno non va conteggiato, si hanno nove bit in mantissa, mentre il massimo esponente è 31 (dato da $2^6 - 1 - k$, con $k = \text{eccesso}$, pari a 32) da cui:

massimo = $(2 - 2^{-9}) * 2^{31}$

Il calcolo della mantissa è stato derivato da:

$$\sum_{i=0}^9 2^{-i} = \frac{1 - 2^{-(9+1)}}{1 - 2^{-1}} = \frac{1 - 2^{-10}}{2^{-1}} = 2 \cdot (1 - 2^{-10}) = 2 - 2^{-9}$$

Soluzione alla domanda 2.

Utilizziamo l'algoritmo delle divisioni successive per la parte intera ottenendo

Quoziente	215	107	53	26	13	6	3	1
Resto	1	1	1	0	1	0	1	1

$215_{10} = 11010111_2$.

Utilizziamo l'algoritmo delle moltiplicazioni successive per la parte frazionaria:

Valore	1.05	0.1	0.2	0.4	0.8	1.6
			Da qui in poi non rappresentabili in mantissa...			

Poiché la mantissa ha solo nove bit, dobbiamo troncare il calcolo dalla parte in corsivo:

$0.525_{10} = 0.100001...._2$.

In sintesi:

$215.525_{10} = 1.101011110 * 2^7_2$.

L'esponente si rappresenta con $7 + 32 = 39 \rightarrow 100111$

S	Mantissa									Esponente					
0	1	0	1	0	1	1	1	1	0	1	0	0	1	1	1

Soluzione alla domanda 3.

Il valore calcolato, per via del troncamento, non rappresenta esattamente il valore dato ma : $215 + 2^{-1} = 215.5$, con una perdita di precisione pari a 0,012% circa rispetto ad esso.

ESERCIZIO 3

Per il circuito e le funzioni circuitali si vedano le dispense del corso. In funzione delle operazioni richieste si hanno i valori seguenti per gli operandi e il riporto in ingresso:

s1	s0	F	A	B	C _{in}
0	0	A'	A'	-1	1
0	1	A-B	A	B'	1
1	0	B-A	A'	B	1
1	1	B'	-1	B'	1

s ₁ s ₀ A _i	00	01	11	10
0	1		1	1
1		1	1	

$$A_i^{\text{new}} = s_0 A_i + \bar{s}_0 \bar{A}_i + s_1 \bar{A}_i$$

s ₁ s ₀ B _i	00	01	11	10
0	1	1	1	
1	1			1

$$B_i^{\text{new}} = s_0 \bar{B}_i + \bar{s}_0 B_i + \bar{s}_1 \bar{s}_0$$

$$c_{\text{in}} = 1$$

L'esercizio per essere completato richiede il disegno su porte logiche dei tre circuiti di cui sopra, la cui realizzazione si lascia allo studente.

ESERCIZIO 4

Soluzione domanda 1.

La durata di un ciclo di clock è pari a $1/2 = 0.5 \text{ ns}$

La lettura su un bus sincrono avviene secondo il protocollo seguente:

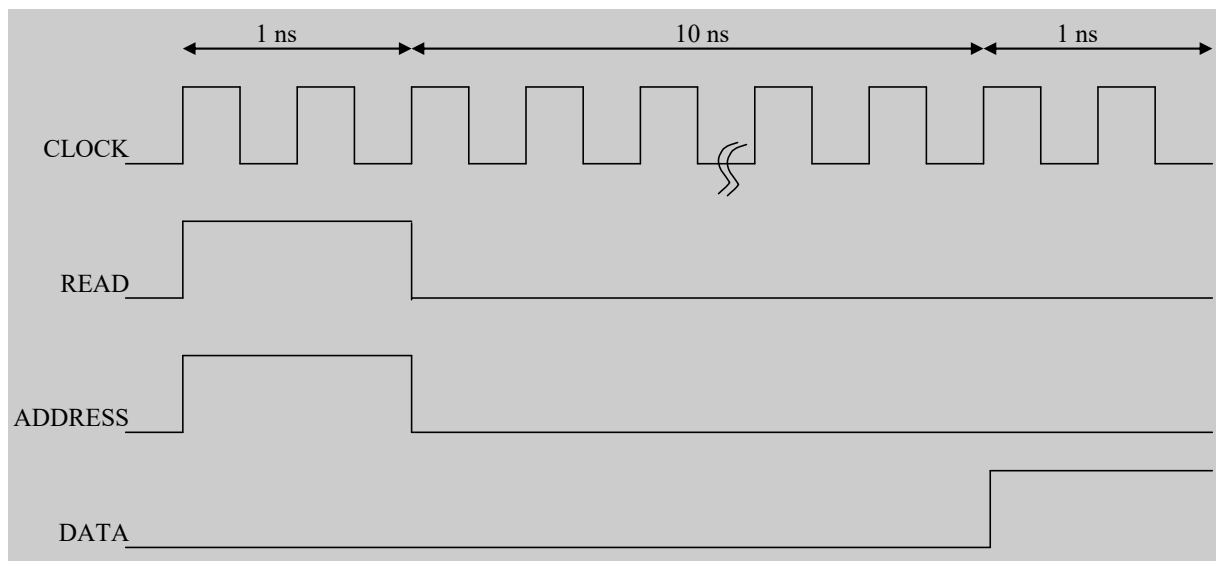
- Segnale di READ sulla linea di controllo e contemporaneamente l'indirizzo della locazione in cui risiede il dato sulla linea indirizzi:

2 cicli di clock = 1 ns

- Lettura della parola dalla memoria: $0.5 \times 20 = 10 \text{ ns}$
- Trasferimento della parola dalla memoria:

trasferimento della parola = 1 ns

Tempo totale per leggere una parola dalla memoria = $(2 + 10) \text{ ns} = 12 \text{ ns}$



Soluzione domanda 2.

Il testo indica chiaramente che si tratta di una stampa di 10 parole. Abbiamo due istruzioni che impiegano ciascuna 1.2 ns per essere eseguite, e poi un ciclo che ripete il trasferimento di dette parole con le seguenti tempistiche $10 \times (12 + 1.2 + 1.2) = 10 \times 14.4 = 144 \text{ ns}$. In totale il trasferimento di queste dieci parole richiede $2.4 + 144 = 146.4 \text{ ns}$, ovvero 14.64 ns a parola. La velocità di trasferimento è dunque pari a $32 \text{ bit} / 14.64 = 2.19 \text{ Gbps} = 0.27 \text{ GB/s}$

Soluzione domanda 3.

Il trasferimento con DMA e furto di ciclo implica che vengono fatti $307200/32$ furti al secondo, ovvero abbiamo un decremento del numero di cicli disponibili al secondo pari a 9600, "rubati" dal DMA. Poiché ogni istruzione richiede 4 cicli di clock, ciò significa che l'istruzione rate sarà ridotto di 2400 istruzioni al secondo (l'istruzione rate nominale è di 500 MIPS).