

Nome, cognome, matricola

Calcolatori Elettronici (12AGA) – esame del 8.9.2015 - A

Domande a risposta chiusa (è necessario rispondere correttamente ad almeno 6 domande).
Non è possibile consultare alcun tipo di materiale. Tempo: 15 minuti.

1	Si consideri l'istruzione ADD VAR, AX dove VAR è una variabile su 16 bit in memoria. Sapendo che il codice macchina dell'istruzione occupa 4 byte, quanti accessi in memoria richiede l'esecuzione dell'istruzione in un sistema 8086?		
2	Si confrontino un sommatore con architettura <i>Ripple Carry</i> e un altro con architettura <i>Carry Lookahead</i> . Quale delle seguenti affermazioni è <u>vera</u> ?	Il sommatore con architettura Carry Lookahead è più veloce di quello con architettura <i>Ripple Carry</i> .	A
		Il sommatore con architettura Carry Lookahead richiede meno hardware di quello con architettura <i>Ripple Carry</i> .	B
		Il sommatore con architettura Carry Lookahead è più facile da progettare di quello con architettura <i>Ripple Carry</i> .	C
		Il sommatore con architettura Carry Lookahead utilizza per ciascun calcolo un numero di colpi di clock minore di quello con architettura <i>Ripple Carry</i> .	D
3	Che cos'è un <i>microcontrollore</i> ?	Un processore destinato specificatamente ad applicazioni robotiche	A
		Un processore con un set ridotto di istruzioni	B
		Un dispositivo destinato ad applicazioni special purpose, che integra sullo stesso dispositivo un processore, alcuni moduli di memoria, e alcune periferiche	C
		Un processore integrato a bordo di un singolo circuito integrato	D
4	Si consideri una cache direct-mapped composta da 512 linee da 32 byte ciascuna che usa la tecnica nota come write-back. Quanti dirty-bit esistono nella cache?	0	A
		1	B
		512	C
		16K	D
5	Dove è memorizzata la <i>Interrupt Vector Table</i> in un sistema a microprocessore?	Nella memoria ROM del sistema	A
		Nella memoria RAM del sistema	B
		Nella cache	C
		Nell'Interrupt Controller	D
6	Si desidera utilizzare la porta A di un 8255 in modo che sia collegata con una connessione bidirezionale a una periferica e scateni una richiesta di interrupt ogni volta che la periferica connessa è pronta a inviare o ricevere un nuovo dato. In quale modo va programmata tale porta?	Modo 0	A
		Modo 1	B
		Modo 2	C
		L'8255 non prevede un simile meccanismo di funzionamento	D
7	Si consideri un banco di memoria composto da 1M parole da 24 bit ciascuna. Quanti moduli composti ciascuno da 256K parole da 8 bit ciascuna sono richiesti per la sua realizzazione?		
8	Quali operazioni esegue l'istruzione IRET?	Ripristina dallo stack il valore di IP (ed eventualmente CS)	A
		Ripristina dallo stack il valore di IP (ed eventualmente CS) e del registro di stato (PSW)	B
		Ripristina dallo stack il valore di IP (ed eventualmente CS), del registro di stato (PSW) e dei registri modificati dalla procedura	C
		Ripristina dallo stack il valore di IP (ed eventualmente CS), del registro di stato (PSW) e dei registri modificati dalla procedura, e segnala all'8259 che è terminata la procedura di servizio dell'interrupt	D
9	Sia dato un vettore di word VET contenente N valori interi con segno (N è una costante predefinita). Si scriva un frammento di codice che calcoli il valore minimo tra gli elementi di VET e lo scriva in AX.		

Risposte corrette

1	2	3	4	5	6	7	8	9
4	A	C	C	B	C	12	B	

Domanda 1

In un sistema 8086 ogni accesso in memoria comporta la lettura o scrittura di 16 bit. Quindi l'istruzione in questione richiede

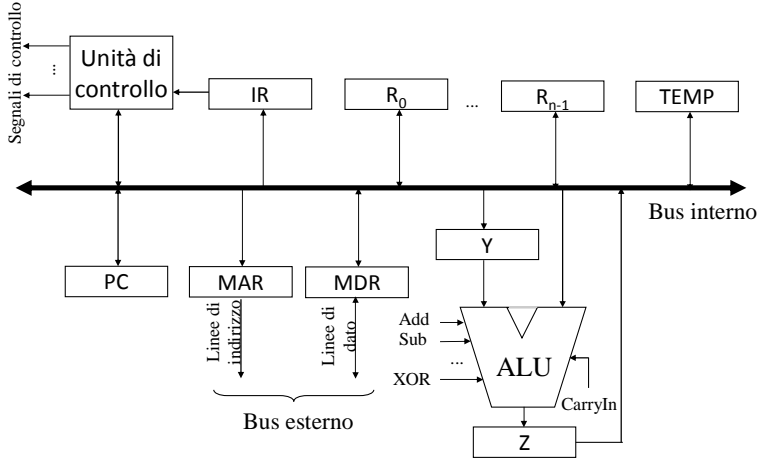
- 2 accessi per il fetch
- 1 accesso per leggere VAR
- 1 accesso per scrivere VAR.

In totale sono necessari 4 accessi.

Domanda 9 (esempio di soluzione)

```
MOV    CX, N
LEA    BX, VET
MOV    AX, 07FFFh    ; valore massimo positivo
Ciclo: CMP    AX, [BX]
        JL     Dopo
        MOV    AX, [BX]
Dopo:  ADD    BX, 2
        LOOP   Ciclo
```

Domande a risposta aperta (sino a 5 punti per ogni domanda) – Non è possibile consultare alcun materiale -
Tempo: 40 minuti.

10	<p>Si scrivano le microistruzioni eseguite da un processore avente l'architettura in figura durante l'esecuzione dell'istruzione ADD R1, [R2], R3</p> <p>Tale istruzione somma il contenuto di R3 al contenuto della cella di memoria il cui indirizzo è scritto in R2, e scrive il risultato in R1.</p>  <p>The diagram illustrates a computer architecture with the following components and connections:</p> <ul style="list-style-type: none">Internal Bus (Bus interno): A central horizontal bus connecting the Unità di controllo (Control Unit), IR (Instruction Register), registers R₀ through R_{n-1}, and the TEMP register.External Bus (Bus esterno): A lower horizontal bus connecting the PC (Program Counter), MAR (Memory Address Register), and MDR (Memory Data Register).Control Unit: Receives Segnali di controllo (control signals) and sends signals to the PC, MAR, MDR, and IR.Registers: R₀ through R_{n-1} are connected to the internal bus. TEMP is also connected to the internal bus.ALU (Arithmetic Logic Unit): Receives data from the internal bus (labeled Y) and the MDR (via Linee di dato). It performs operations like Add, Sub, and XOR. It has a CarryIn input and a Z (Zero) output.Other Units: PC sends Linee di indirizzo (address lines) to the MAR. The MAR and MDR are connected to the external bus.
11	<p>Si illustri brevemente il meccanismo di arbitraggio centralizzato di un bus noto come <i>daisy chaining</i>, corredando la propria risposta di un disegno esplicativo delle connessioni necessarie. Si utilizzi come esempio il caso in cui i dispositivi connessi siano 5, e si riportino i segnali necessari in tale situazione.</p>

12	Si disegni l'architettura di un Flip Flop di tipo D in termini di porte logiche e se ne illustri il funzionamento utilizzando la relativa tabella degli stati.
13	Si descrivano le operazioni eseguite da un sistema a processore dotato di DMA Controller (DMAC) a partire dal momento in cui il processore decide di eseguire un trasferimento da memoria a periferico utilizzando il DMAC sino al momento in cui l'intera procedura di trasferimento è conclusa.

Esercizio di programmazione

sino a 12 punti – è possibile consultare qualunque materiale cartaceo - tempo: 60 minuti

Sia dato un vettore *vett_input* di interi con segno su 16 bit di dimensione N (N dichiarato come costante), che rappresenta una funzione definita in N punti. Si vuole interpolare linearmente ogni coppia (*vett_input*[j], *vett_input*[j+1]), ossia si vuole calcolare il valor medio di ogni coppia di punti.

Si scriva una procedura **interpola** che riceve tramite stack

- l'offset del vettore in input *vett_input*
- il numero di elementi N
- l'offset del vettore in output *vett_output*.

vett_output è un vettore di interi con segno su 16 bit di dimensione $2 * N - 1$, contenente sia i valori di ingresso sia quelli interpolati. Più precisamente:

$$vett_output[i] = \begin{cases} vett_input\left[\frac{i}{2}\right] & \text{se } i \text{ è pari} \\ \frac{vett_input\left[\frac{i-1}{2}\right] + vett_input\left[\frac{i+1}{2}\right]}{2} & \text{se } i \text{ è dispari} \end{cases}$$

Si noti che il valore medio fra ogni coppia di elementi di *vett_input* è sicuramente rappresentabile su 16 bit, ma ciò non vale a priori per la somma di due elementi consecutivi di *vett_input*. È necessario quindi effettuare una scelta opportuna dell'ordine dei calcoli e della dimensione dei dati intermedi in modo che non si verifichi overflow nei calcoli.

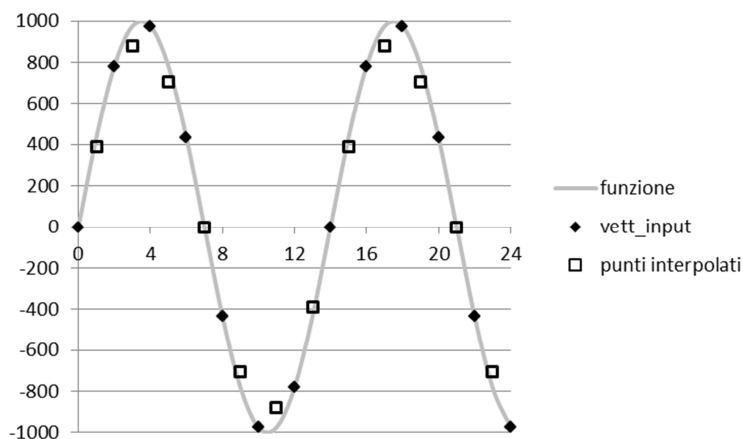
Di seguito un esempio di programma chiamante:

```
N EQU 13
.MODEL small
.STACK
.DATA
vett_input    DW 0, 782, 975, 434, -434, -975, -782
              DW 0, 782, 975, 434, -434, -975
vett_output   DW 2 * N - 1 DUP (?)

.CODE
.STARTUP
[...]
```

PUSH OFFSET vett_input
PUSH N
PUSH OFFSET vett_output
CALL interpola
[...]
.EXIT

Di seguito si fornisce una rappresentazione grafica con i valori di *vett_input* indicati nel codice:



vett_output contiene i valori di *vett_input* alternati con i valori interpolati. Nell'esempio riportato, dopo la chiamata della funzione **interpola**, *vett_output* = [0, 391, 782, 878, 975, 704, 434, 0, -434, -705, -975, -879, -782, -391, 0, 391, 782, 878, 975, 704, 434, 0, -434, -705, -975].