

Compito di Architettura dei Calcolatori - A.A. 2010-11  
 Prova di esame del 13 giugno 2011

COGNOME:

NOME:

MATRICOLA:

**Istruzioni:** Spiegare chiaramente TUTTE le assunzioni che vengono effettuate per chiarire eventuali punti che si ritengono ambigui o non specificati.

1) [10 punti] Illustrare i principi generali ed il flusso di controllo per la gestione di una "pipeline" per ottimizzare le prestazioni della CPU, discutendo sia il caso di salto incondizionato che di salto condizionato.

**SVOLGIMENTO:**

Si veda il cap.12 del libro di testo e dei lucidi presentati a lezione, in particolare i lucidi da 38 a 38 (compresi).

2) [10 punti] Siano date due serie di N valori positivi in notazione *non complementata*  $a(i)$  e  $c(i)$  contenute in N celle di memoria consecutive a partire da quelle di indirizzo A e C. Dapprima analizzare e descrivere le possibili differenti relazioni sussistenti tra 2 valori  $a(i)$ ,  $c(i)$ . Poi, scrivere un programma nel linguaggio Assembly di Virtual CPU che scrive nella cella  $r(i)$  un diverso intero positivo per ognuna di tali relazioni. I valori  $r(i)$  sono scritti in celle di memoria consecutive a partire da quella di indirizzo R.

I valori A, C, R, N sono contenuti nelle celle di memoria di indirizzo rispettivamente da 1 a 4. Assumere che i valori A, C, R, N sono tali da non creare mai problemi di sovrapposizione tra le celle di memoria

Commentare con adeguato dettaglio la logica seguita e le istruzioni usate

**SVOLGIMENTO:**

Sono possibili tre distinte relazioni tra  $a(i)$  e  $c(i)$ :

- $a(i) < c(i)$ , nel qual caso scriviamo 1 in  $r(i)$ ,
- $a(i) = c(i)$ , nel qual caso scriviamo 2 in  $r(i)$ ,
- $a(i) > c(i)$ , nel qual caso scriviamo 3 in  $r(i)$ ,

Assumiamo N sia sempre maggiore di 0.

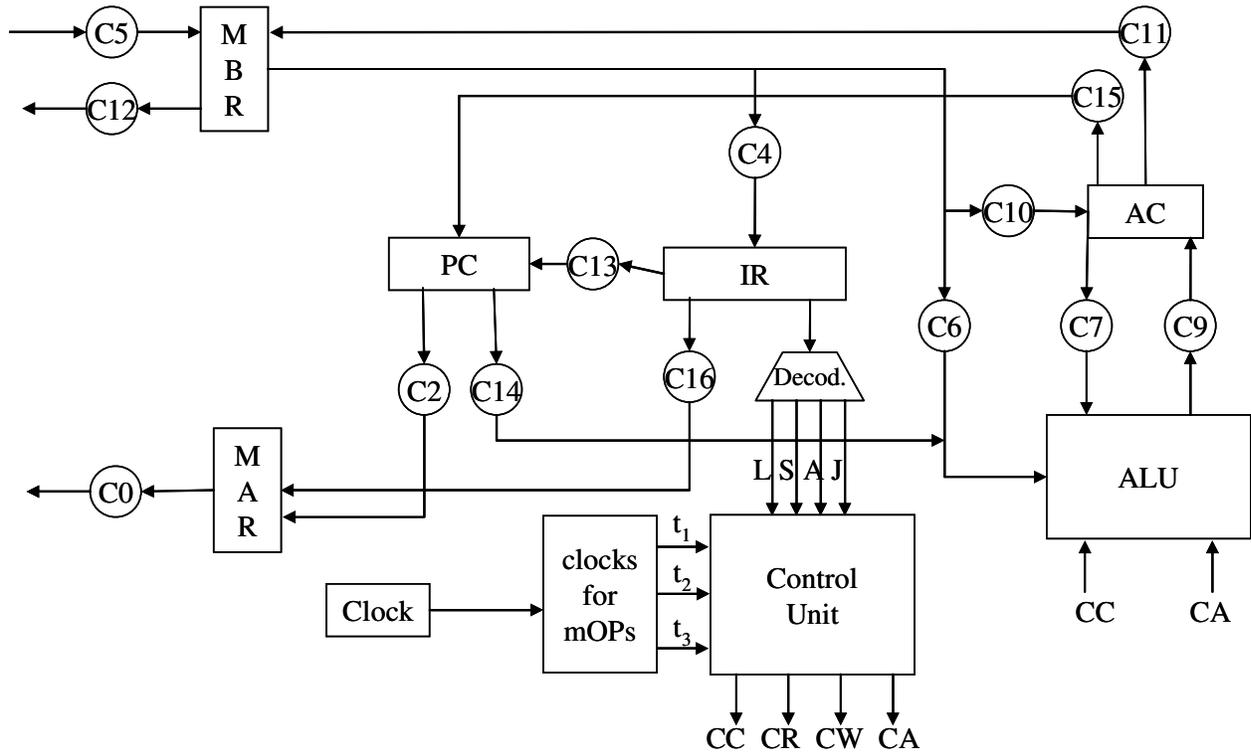
```

0  JMP 5          ; salta alla cella d'inizio del programma
; I DATI del programma, memorizzati in celle che sono usate come variabili di ingresso al programma
1  ; contiene l'indirizzo del valore di a(i), cioè l'indirizzo di a(1) all'inizio
2  ; contiene l'indirizzo del valore di c(i), cioè l'indirizzo di c(1) all'inizio
3  ; contiene l'indirizzo del valore di r(i), cioè l'indirizzo di r(1) all'inizio
4  ; contiene il numero di valori da confrontare
; IL PROGRAMMA
; inizializzazione con il calcolo del primo valore F_3
5  LOAD @@1      ; si carica nell'accumulatore a(i)
6  SUB @@2       ; gli si sottrarre c(i)
7  JL 17         ; se a(i)<c(i) si gestisce questo caso
8  JZ 20         ; se a(i)=c(i) si gestisce questo caso
9  JG 23         ; se a(i)>c(i) si gestisce questo caso
10 DEC @4       ; si decrementa il numero di valori da confrontare
11 JZ 16         ; se non ce ne sono più si salta all'arresto altrimenti si prosegue
12 INC @1       ; si incrementa l'indirizzo di a(i) per il passo successivo
13 INC @2       ; si incrementa l'indirizzo di c(i) per il passo successivo
14 INC @3       ; si incrementa l'indirizzo di r(i) per il passo successivo
15 JMP 5        ; si salta a ricominciare il passo successivo
16 HLT
; caso a(i)<c(i)
17 LOAD 1       ; si carica 1 nell'accumulatore
18 STORE @@3    ; e lo si scrive in r(i)
19 JMP 10       ; si continua per l'eventuale passo successivo
; caso a(i)=c(i)
20 LOAD 2       ; si carica 2 nell'accumulatore
21 STORE @@3    ; e lo si scrive in r(i)
22 JMP 10       ; si continua per l'eventuale passo successivo
; caso a(i)>c(i)
23 LOAD 3       ; si carica 3 nell'accumulatore
24 STORE @@3    ; e lo si scrive in r(i)
25 JMP 10       ; si continua per l'eventuale passo successivo

```

NB: Nel simulatore ENIAC l'implementazione di JA e JB non è corretta. Per questo motivo nella soluzione su esposta sono state usate JG e JL che assumono di avere valori rappresentati in notazione complementata. La soluzione presentata è pertanto valida in queste ipotesi.

3) [10 punti] Dato lo schema della semplicissima CPU (VS0) sotto disegnato, fornire e spiegare le micro-istruzioni della micro-procedura usata da un'unità di controllo micro-programmata per l'effettuazione della fase di prelevamento della prossima istruzione (fetch), nell'ipotesi che la struttura della parola di controllo (Control Word) contenga UN SOLO flag di salto e contenga due campi con l'indirizzo della successiva micro-istruzione, descrivendo e spiegando inoltre eventuali circuiteria aggiuntiva presente nell'unità di controllo.



SVOLGIMENTO:

Si veda il cap.17 del libro di testo e dei lucidi presentati a lezione, in particolare i lucidi 6 e 17.