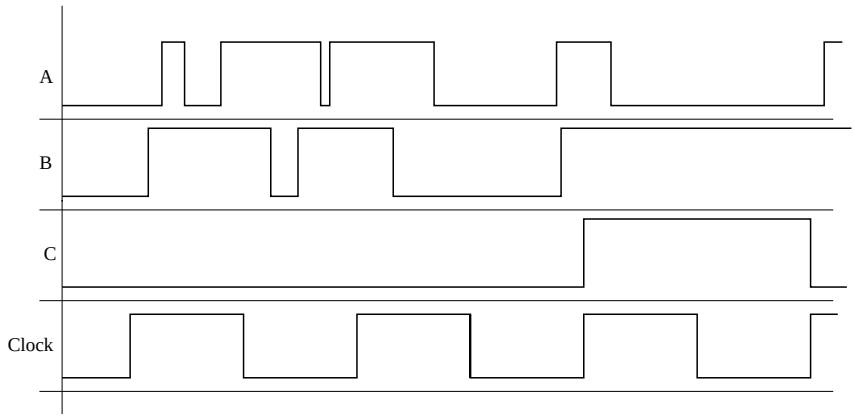
Corso di Architettura degli Elaboratori
Prova scritta del 22 Febbraio 2008
Tempo a disposizione: ore 1:50.

Si ricorda che:

- Per quanto possibile, occorre scrivere in bella calligrafia (il testo illeggibile non verrà preso in considerazione).
- Su tutti i fogli che vi abbiamo consegnato occorre riportare cognome, nome e numero di matricola.
- Occorre riportare in modo chiaro tutti i passi che portano alla determinazione del risultato.
- Il numero dell'esercizio che si sta svolgendo va sempre riportato in modo chiaro.
- Non è consentita la consultazione di appunti, libri, etc.
- Non è consentito l'uso di calcolatrici, telefoni cellulari, etc.
- Non è concesso chiedere alcunché ai docenti e agli altri studenti.
- Occorre consegnare anche la brutta copia ai docenti.

Esercizio 1. *(Punti 3, la risposta occupi al massimo 5 righe)*
Si parli della Legge di Moore.

Esercizio 2. *(Punti 10)*
Si vuole costruire un circuito sequenziale sincrono con due entrate A e B e una singola uscita C. Il valore di C deve essere 1 se e solo se gli ultimi *tre* valori binari letti su A (in corrispondenza, quindi, degli ultimi tre fronti di salita) sono uguali agli ultimi tre valori binari letti su B. In tutti gli casi C vale 0. In altre parole, il circuito sequenziale riconosce l'uguaglianza tra la sequenza di tre bit letta in A e la sequenza di tre bit letta in B. Ad esempio:



Si proceda come segue:

a) Si costruisca un diagramma di stato per il circurito sequenziale sincrono sopra descritto.
b) Si costruisca la relativa tabella di stato.
c) Si costruiscano le equazioni di ingresso ai flip-flop (che si suppongono di tipo D).
d) Si semplifichino le equazioni di ingresso di cui al punto precedente utilizzando le mappe di Karnaugh.

e) Si costruisca il circuito sequenziale vero e proprio.

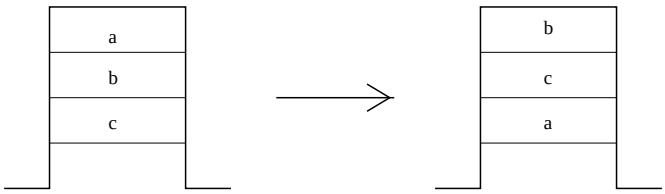
Esercizio 3. *(Punti 3, la risposta occupi al massimo 10 righe)*
Si parli brevemente della classificazione dei circuiti integrati in base al numero di porte logiche in essi contenute.

Esercizio 4. *(Punti 5)*
Si scriva una subroutine nel linguaggio IJVM che prenda in ingresso un unico parametro un intero a (che si suppone non essere negativo) e restituisce la somma dei primi a interi, ossia $1 + 2 + 3 + 4 + \ldots + (a - 1) + a$.
Suggerimento: un programma che risolve lo stesso problema, ma scritto in un linguaggio di alto livello (quindi non in IJVM) potrebbe essere il seguente:

```
int foo(int a)
{
 if (a==0)
   i=0;
 else
   i=a+foo(a-1);
 return i;
}
```

Un modo per risolvere il problema, consiste quindi nel tradurre il programma di cui sopra in un equivalente programma IJVM. In alternativa, si potrebbe anche scrivere il programma IJVM direttamente, senza partire da alcun programma di più alto livello.

Esercizio 5. *(Punti 5)*
L'istruzione `ROD` non esiste in IJVM. L'istruzione `ROD` ha l'effetto di cambiare l'ordine delle ultime tre parole in cima allo stack, come descritto nella figura seguente:



Si estenda il microcodice di Mic-1 con una sequenza di microistruzioni per `ROD`.

Esercizio 6. *(Punti 4, la risposta occupi al massimo 12 righe)*
Si spieghi chiaramente in che senso la modalità di indirizzamento a registro, la modalità di indirizzamente a registro indiretto e l'indirizzamento indicizzato sono diverse tra loro. Sono disponibili nel linguaggio assembly del processore 8088? Si diano alcuni esempi significativi.