

**Calcolatori Elettronici, A.A. 2003-2004**

**13 aprile 2004**

**(Compito A\_svolgimento)**

- La durata della prova è di 2h:30m.
- E' vietato consultare libri, eserciziari, appunti ecc.. Chiunque venga trovato in possesso di documentazione relativa al corso vedrà annullato il proprio compito.
- Riportare nell'intestazione di ogni foglio (a stampatello) i seguenti dati: Cognome, Nome, Matricola, Compito.

**Esercizio 1 [15%]**

Data la seguente funzione booleana avente 4 ingressi ed una uscita:

On-Set={0,4,5,6,7,14}, DC-Set={2,8,10,13,15}

1. Utilizzando il metodo delle mappe di Karnaugh sintetizzarla in forma minima in prima forma canonica. [45%]

cd\ab	00	01	11	10
00	1	1		-
01		1	-	
11		1	-	
10	-	1	1	-

$$f(a,b,c,d) = \underline{a}b + c\underline{d} + \underline{b}d.$$

2. Utilizzando il metodo delle mappe di Karnaugh sintetizzarla in forma minima in seconda forma canonica. [45%]

cd\ab	00	01	11	10
00			0	-
01	0		-	0
11	0		-	0
10	-			-

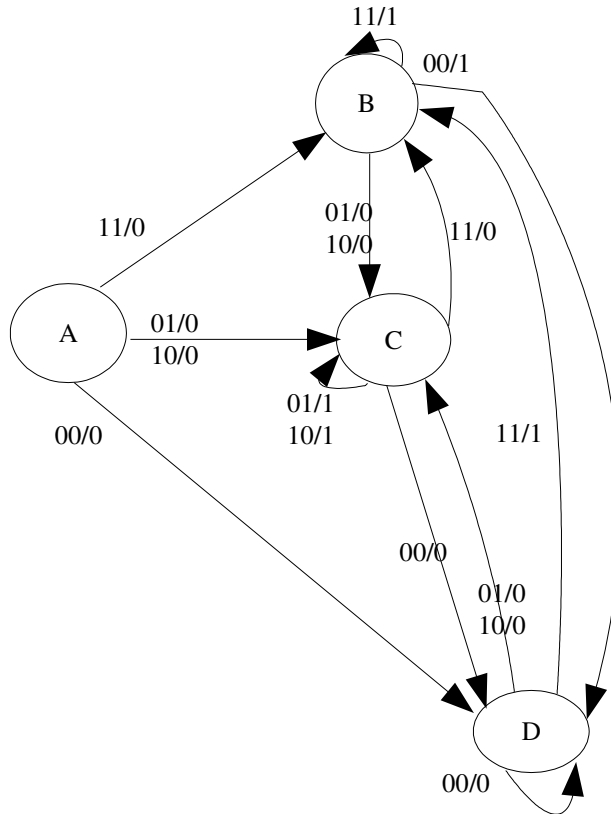
$$F(a,b,c,d) = (\underline{a} + c)(b + \underline{d}).$$

3. Dire quali delle due realizzazioni (prima o seconda forma canonica) è la meno costosa in termini di letterali. [10%]

**Esercizio 2 [30%]**

Progettare una rete sequenziale a due ingressi a e b ed una uscita z tale che l'uscita è 1 se e solo se gli ultimi due caratteri sulla sequenza a e gli ultimi due caratteri sulla sequenza b contengono complessivamente un numero pari di 1 (ovvero  $z(t)=1$  se la stringa  $a(t-1),a(t),b(t-1),b(t)$  contiene un numero pari di "1").

1. Tracciare il diagramma a stati. [50%]



2. La tabella degli stati è:

	<b>00</b>	<b>01</b>	<b>11</b>	<b>10</b>
<b>A</b>	D/0	C/0	B/0	C/0
<b>B</b>	D/1	C/0	B/1	C/0
<b>C</b>	D/0	C/1	B/0	C/1
<b>D</b>	D/0	C/0	B/1	C/0

3. Minimizzare il diagramma degli stati. [10%]

Il diagramma è già in forma minima.

UNIVERSITÀ DEGLI STUDI DI CATANIA, FACOLTÀ DI INGEGNERIA  
CORSO DI LAUREA IN INGEGNERIA ELETTRONICA

4. Sintetizzare le funzioni di stato prossimo e di uscita utilizzando flip-flop di tipo JK. [30%]

Assegnazione degli stati: A=00, B=01, C=11, D=10. La tabella delle transizioni è quindi:

Q1Q0\ab	00	01	11	10
00	11/0	10/0	01/1	10/0
01	11/1	10/0	01/1	10/0
11	11/0	10/1	01/0	10/1
10	11/0	10/0	01/0	10/0

La funzione d'uscita è:

$$z(a,bQ1,Q0) = \overline{a}\overline{b}Q1Q0 + a\overline{b}Q1Q0 + \overline{a}bQ1Q0 + \overline{a}\overline{b}Q1Q0 + \overline{a}bQ1Q0$$

La tabella delle eccitazioni in riferimento all'utilizzo di FF J-K è:

Q1Q0\ab	00	01	11	10
00	1-,1-	1-,0-	0-,1-	1-,0-
01	1-, - 0.	1-, - 1	0-, - 0	1-, - 1
11	- 0, - 0	- 0, - 1	- 1, - 0	- 0, - 1
10	- 0,1-	- 0,0-	- 1,1-	- 0,0-

Si trova:

$$J1 = \overline{a}Q1 + ab \quad ; \quad K1 = ab$$

$$J2 = \overline{a}b + ab = a \oplus b \quad ; \quad K2 = \overline{a}b + ab = a \oplus b$$

5. Tracciare lo schema della rete. [10%]

E' banale.

### Esercizio 3 [15%]

Descrivere in che modo può essere sfruttata la località spaziale e temporale dei riferimenti esibita da un programma (non più di una pagina compresi eventuali schemi).

Vedi appunti e libro.

### Esercizio 4 [10%]

Un programma viene compilato su due sistemi basati rispettivamente su architettura RISC e CISC. Il sistema basato su architettura RISC ha una frequenza di clock doppia rispetto al sistema basato su architettura CISC ed esegue mediamente 0.8 istruzioni per ogni ciclo di clock a differenza di quello basato su CISC che ne esegue 0.5. Il numero di istruzioni eseguite dal CISC è il 60% di quelle eseguite dal RISC. Quali dei due sistemi è più veloce?

$$TCPU_{risc} = IC_{risc} * CPI_{risc} * T_{risc}$$

$$TCPU_{cisc} = IC_{cisc} * CPI_{cisc} * T_{cisc}$$

$$IC_{cisc} = 0.6 IC_{risc}, T_{risc} = 2 * T_{cisc}, CPI_{cisc} = 1/0.5, CPI_{risc} = 1/0.8$$

$TCPU_{risc}/TCPU_{cisc} < 1$  quindi il sistema basato su RISC è più veloce di quello basato su CISC sebbene quest'ultimo esegue meno istruzioni.

UNIVERSITÀ DEGLI STUDI DI CATANIA, FACOLTÀ DI INGEGNERIA  
CORSO DI LAUREA IN INGEGNERIA ELETTRONICA

**Esercizio 5 [30%]**

Tradurre in assembly del DLX il seguente programma pseudo-C (è possibile utilizzare, senza implementarla, la funzione InputUnsigned discussa a lezione).

```
#define N 10
main()
{
    int vect[N];
    int i, imin;

    <Acquisizione del vettore da tastiera>
    imin = 0;
    i = 1;
    while (i < N)
    {
        if (vect[i] < vect[imin])
            imin = i;
        i++;
    }
    printf("Il minimo è %d e si trova in posizione %d\n", vect[imin], imin);
}

.data
vect:      .space 10*4
prompt:    .ascii "Inserisci un numero positivo: "
str:       .ascii "Il minimo è %d e si trova in posizione %d\n"
.align 2
printfpar: .word stringa
vimin:     .space 4
imin:      .space 4

.code
; Acquisizione dati
addi r2, r0, 10
addi r3, r0, 0
loop:
addi r1, r0, prompt
jal  InputUnsigned
sw   vect(r3), r1
addi r3, r3, 4
subi r2, r2, 1
bnez loop

; Ricerca dell'indice del vettore il
; cui elemento è minimo
addi r1, r0, 0 ; imin -> r1
addi r2, r0, 4 ; i -> r2
addi r3, r0, 9 ; counter -> r3
while:
bez  r3, end_while
lw   r4, vect(r2) ; v[i] -> r4
lw   r5, vect(r1) ; v[imin] -> r5
sl   r6, r5, r4
bnez r6, update
j    next_iterazion
update:
addi r1, r0, r2
next_iterazion:
subi r3, r3, 1
addi r2, r0, 4
j    while

; Visualizzazione
lw   r2, vect(r1)
sw   vimin, r2
shri r1, r1, 2
sw   imin, r1
addi r14, r0, printfpar
trap 5

trap 0
```