

PROVA SCRITTA DEL MODULO DI
CALCOLATORI ELETTRONICI
CORSI DI LAUREA IN
INGEGNERIA BIOMEDICA
INGEGNERIA ELETTRICA, ELETTRONICA ED INFORMATICA
17 luglio 2018

NOME:

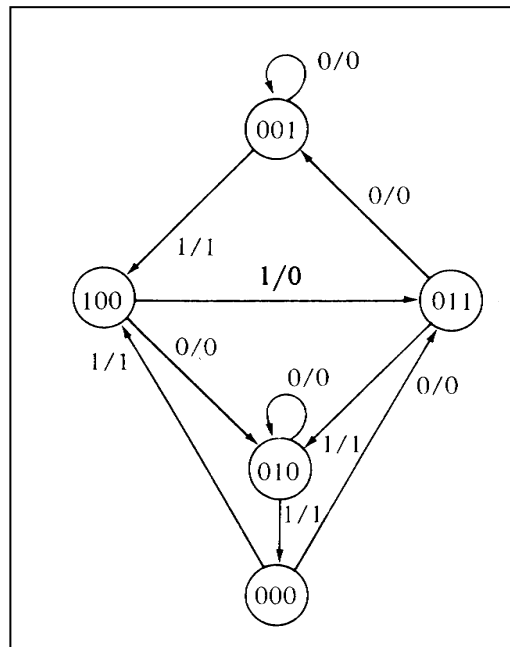
COGNOME:

MATRICOLA:

ESERCIZIO 1 (8 punti)

Un circuito sequenziale presenta un ingresso, una uscita e può essere rappresentato con il grafo degli stati in figura. Si richiede di progettare la corrispondente rete sequenziale utilizzando FF-JK.

Calcolare le forme minime per le variabili di eccitazione dei flip flop e l'uscita usando le mappe di Karnaugh.



ESERCIZIO 2 (4 punti)

I trasferimenti di parole a/dalla memoria di un calcolatore sono codificate utilizzando il codice di Hamming. Si consideri la stringa di 12 bit 011001111110 (il bit meno significativo è a sinistra), risultato della codifica di una parola di N bit secondo il codice di Hamming.

1. (1 punto) Calcolare N, supponendo di aver fatto uso del numero minimo di bit di controllo necessario per una stringa di 12 bit;
2. (3 punti) indicare eventuali errori nella stringa codificata, specificando quale dei bit è stato alterato.

ESERCIZIO 3 (8 punti)

Implementare una procedura in Assembly MIPS che dati l'indirizzo iniziale di un vettore v in \$4 e la sua dimensione N in \$5, restituisca il minimo valore contenuto nel vettore in \$6.

ESERCIZIO 4 (6 punti)

Si consideri una memoria primaria costituita da 32 parole indirizzabili e da una cache formata da quattro parole.

1. (2 punti) Spiegare come vengono interpretati gli indirizzi logici per recuperare l'informazione contenuta nella cache nel caso venga usata la modalità di indirizzamento diretto con blocchi di due parole.
2. (4 punti) Ipotizzando la cache vuota all'istante iniziale, indicare lo stato finale della cache e il numero di hit nel caso vengano inoltrate le seguenti chiamate (indirizzi espressi in decimale, primo indirizzo 0): 4, 6, 6, 5, 1, 3, 7, 6, 3, 6.

ESERCIZIO 5 (7 punti)

Si consideri un calcolatore in cui la CPU esegue 10^5 istruzioni/s. L'esecuzione di una istruzione richiede 5 cicli di clock, 3 dei quali tengono occupato il bus di sistema. Si ipotizzi che il 75% dell'Instruction Rate sia usato dalla CPU per eseguire programmi che non contengono trasferimenti di I/O. L'ampiezza della linea dati del bus è pari a 32 bit.

1. (4 punti) Calcolare la massima frequenza di trasferimento dati ottenibile (espressa in kB/s) nel caso in cui si usi la modalità "transparent" DMA. Si ipotizzi che una operazione di lettura/scrittura della memoria richieda un ciclo di clock e che una parola sia pari a 32 bit.
2. (3 punti) Si descrivano in modo chiaro e sintetico le differenze tra il sistema di arbitraggio centralizzato e quello distribuito, facendo particolare riferimento ai sistemi di gestione delle periferiche "daisy chain" e "a richieste indipendenti".

ESERCIZIO 1

Soluzione

Per risolvere l'esercizio è sufficiente sapere scrivere le tabelle delle transizioni facilmente ricavabili dal grafo degli stati.

Innanzitutto scriviamo la tabella di eccitazione dei due flip flop richiesti, il JK e il T:

*Tabella di eccitazione
di un flip-flop JK*

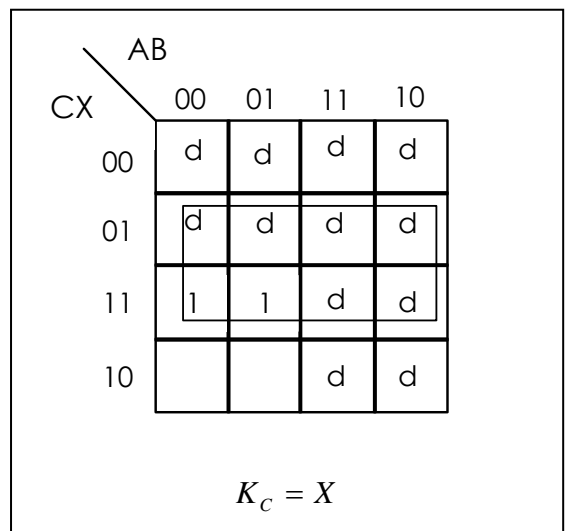
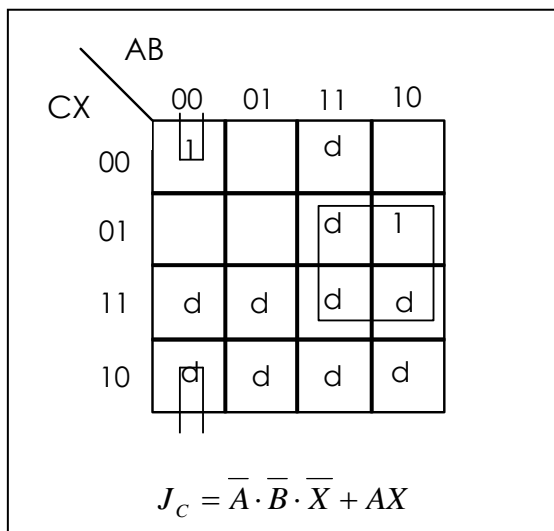
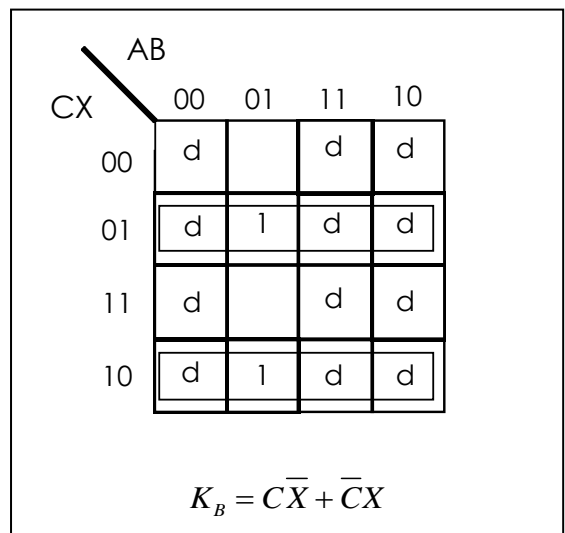
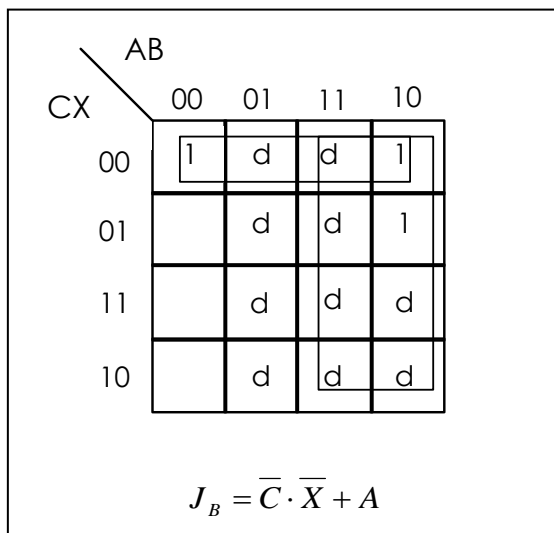
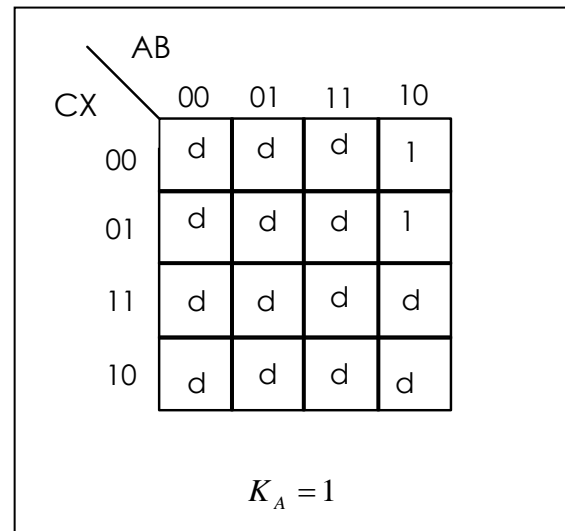
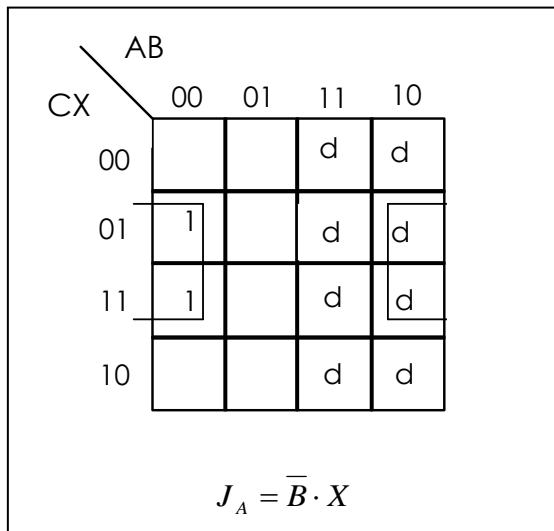
$Q(t)$	$Q(t+1)$	J	K
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

Indicando con A, B, C le variabili di stato, con X l'ingresso e con Z l'uscita, la relativa tabella delle transizioni è:

A	B	C	A'	J _A	K _A	B'	J _B	K _B	C'	J _C	K _C	Z
X=0												
0	0	0	0	0	D	1	1	D	1	1	D	0
0	0	1	0	0	D	0	0	D	1	D	0	0
0	1	0	0	0	D	1	D	0	0	0	D	0
0	1	1	0	0	D	0	D	1	1	D	0	0
1	0	0	0	D	1	1	1	D	0	0	D	0
1	0	1	D	D	D	D	D	D	D	D	D	0
1	1	0	D	D	D	D	D	D	D	D	D	0
1	1	1	D	D	D	D	D	D	D	D	D	0
X=1												
0	0	0	1	1	D	0	0	D	0	0	D	1
0	0	1	1	1	D	0	0	D	0	D	1	1
0	1	0	0	0	D	0	D	1	0	0	D	1
0	1	1	0	0	D	1	D	0	0	D	1	1
1	0	0	0	D	1	1	1	D	1	1	D	0
1	0	1	D	D	D	D	D	D	D	D	D	0
1	1	0	D	D	D	D	D	D	D	D	D	0
1	1	1	D	D	D	D	D	D	D	D	D	0

Si noti che nel caso dell'uscita non è stato usato il *don't care*, per rispettare il vincolo di progetto richiedente che tale variabile assuma il valore 1 solo dove indicato nel grafo degli stati.

Possiamo ora disegnare le mappe di Karnaugh:



		AB			
CX		00	01	11	10
	00				
	01	1	1		
	11	1	1		
	10				

$Z = X \bar{A}$

ESERCIZIO 2

Soluzione

1. Deve essere rispettata la condizione:

$$2^K \geq N + K + 1 \quad (1),$$

dove K è il numero di bit di controllo inseriti. Essendo $N + K = 12$, si evince dalla (1) che il numero minimo di bit di controllo richiesto è 4. Da cui $N = 8$.

2. Nella codifica di Hamming, la sequenza in ingresso presenta la seguente struttura:

c_0	c_1	b_0	c_2	b_1	b_2	b_3	c_3	b_4	b_5	b_6	b_7
0	1	1	0	0	1	1	1	1	1	1	0

Dove $c_0 \dots c_3$ sono i quattro bit costituenti il vettore di controllo, e $b_0 \dots b_7$ gli otto bit trasmessi. La sequenza ricevuta è 10111110.

Per verificare la presenza di un errore, dobbiamo ricalcolare il vettore di controllo a partire dalla sequenza ricevuta. Si ha:

$$c'_0 = b_0 \oplus b_1 \oplus b_3 \oplus b_4 \oplus b_6 = 0$$

$$c'_1 = b_0 \oplus b_2 \oplus b_3 \oplus b_5 \oplus b_6 = 1$$

$$c'_2 = b_1 \oplus b_2 \oplus b_3 \oplus b_7 = 0$$

$$c'_3 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 = 1$$

Il passo successivo è calcolare il vettore di errore dato dalla differenza dei vettori di controllo c e c' (ricordiamo che somma e differenza tra bit producono lo stesso risultato):

$$e_0 = c_0 \oplus c'_0 = 0$$

$$e_1 = c_1 \oplus c'_1 = 0$$

$$e_2 = c_2 \oplus c'_2 = 0$$

$$e_3 = c_3 \oplus c'_3 = 0$$

Poiché il vettore risultante è nullo, non vi sono errori nella stringa di 12 bit data.

ESERCIZIO 3

Soluzione

$\$4 \leftarrow v[0]; \$5 \leftarrow N$

```
max:      addi $29, $29, -16
          sw $4, 0($29)
          sw $8, 4($29)
          sw $9, 8($29)
          sw $10, 12($29)
          lw $6, 0($4)      #assumo che min <- v[0]
          addi $8, $0, 1    #inizializza i=1
          addi $4, $4, 4    #calcola in $4 <- &v[1]=&v[0]+4
for:      beq $8, $5, exit  #i==N, exit
          lw $9, 0($4)      #carico v[i] in $9
          slt $10, $9, $6   #$9 < $6 ->$10
          bne $10, $0, min  #se $10 != 0, aggiorno il min
          j inc             #altrimenti salto a inc
min:      move $6, $9
inc:      addi $8, $8, 1    #i++
          addi $4, $4, 4    #&v[i]++
          j for
exit:     lw $4, 0($29)
          lw $8, 4($29)
          lw $9, 8($29)
          lw $10, 12($29)
          addi $29, $29, 16
          jr $31
```

ESERCIZIO 4

Soluzione.

1.

Memoria indirizzabile: 32 parole = 2^5 parole \rightarrow 5 bit di indirizzamento

<TAG 3 bit> <Cache Index 1 bit> <Offset 1 bit>

2. Nel seguito, è disegnata la cache di quattro parole, e il suo stato finale. Con linee tratteggiate indichiamo la separazione tra un blocco e il successivo, con linee continue la separazione tra due insiemi.

0
1
6
7

Tale stato finale è dovuto alle seguenti dinamiche. La prima riga in tabella riporta la parola chiamata, l'ultima il verificarsi di un hit.

4	6	6	5	1	3	7	6	3	6
4	4	4	4	0	0	0	0	0	0
5	5	5	5	1	1	1	1	1	1
	6	6	6	6	2	6	6	2	6
	7	7	7	7	3	7	7	3	7
		h	h				h		

Hit ratio: 3/10

ESERCIZIO 5

Soluzione

1. Nel caso di 'trasparent DMA' posso trasferire i dati tutte le volte che il bus di sistema è libero. Nel caso in esame questo tempo è pari alla somma del 25% del tempo lasciato libero dall'esecuzione di istruzioni che non coinvolgono I/O, più i due cicli/istruzione in cui il bus è libero. Pertanto durante il 75% del tempo posso trasferire una parola/istr.:

$$0.75 \times 2 \text{ parole/istr} \times 10^5 \text{ istr./s} = 1.5 \times 10^5 \text{ parole/s}$$

Nel restante 25% del tempo posso trasferire 5 parole/istr.:

$$0.25 \times 5 \text{ parole/istr.} \times 10^5 \text{ istr./s} = 1.25 \times 10^5 \text{ parole/s}$$

In **totale**, nel caso di trasferimento con DMA la velocità totale di trasferimento è pari a: **(1.5 + 1.25) $\times 10^5$ parole/s = 2.75 $\times 10^5$ parole/s = 1074.22 kB/s**

2. Si vedano le dispense del corso.