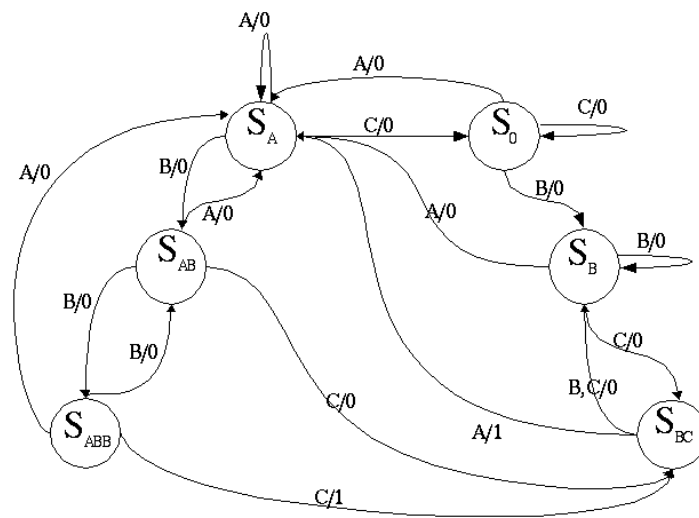


## Soluzioni compito 22.07.2003 - Sistemi di Elaborazione

### Esercizio 1

La soluzione dell'esercizio 1 è riportata in figura 1. Nella stessa figura, oltre al grafo è riportata anche la tabella di flusso.



(a)

	ingressi		
	A	B	C
$S_0$	$S_A/0$	$S_B/0$	$S_0/0$
$S_B$	$S_A/0$	$S_B/0$	$S_{BC}/0$
$S_{BC}$	$S_A/1$	$S_B/0$	$S_B/0$
$S_A$	$S_A/0$	$S_{AB}/0$	$S_0/0$
$S_{AB}$	$S_A/0$	$S_{ABB}/0$	$S_{BC}/0$
$S_{ABB}$	$S_A/0$	$S_{AB}/0$	$S_{BC}/1$

(b)

Figura 1: Soluzione esercizio 1

### Esercizio 2

Per la struttura della ROM, che memorizza un valore per ogni possibile configurazione degli ingressi, è necessario ottenere la tabella di verità delle funzioni  $Z1$  e  $Z2$  da cui si ricava la realizzazione del circuito (fig. 2).

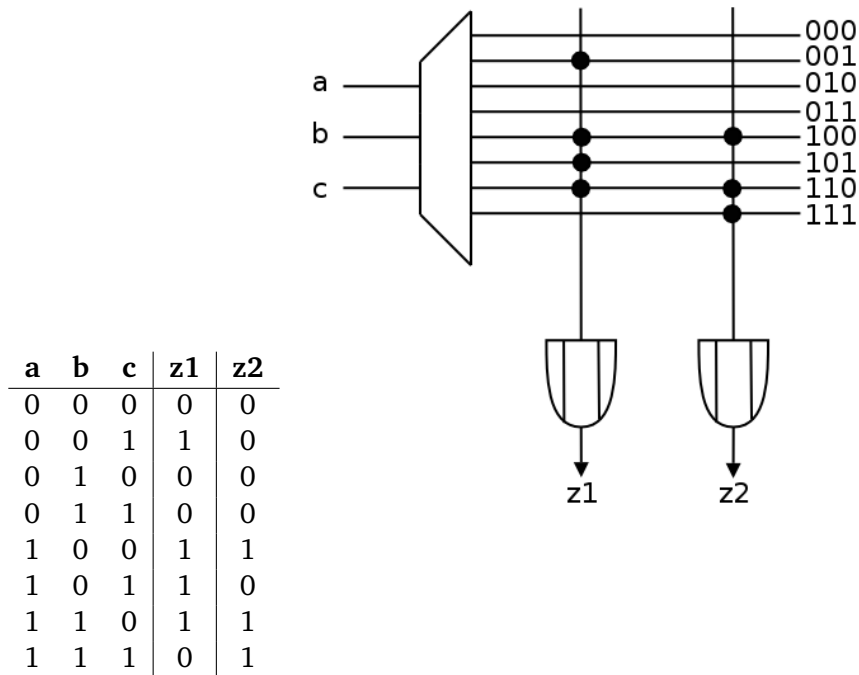


Figura 2: Esercizio 2

### Esercizio 3

Una possibile soluzione è la seguente:

- 0:  $A \rightarrow A, B \rightarrow B, \bar{N} + 2 \rightarrow C, M \rightarrow D, 1 \rightarrow E, 1$
- 1:  $O_0, (K = 0) 2 (K = 1) 8$
- 2:  $C + 1 \rightarrow C, ts(E) \rightarrow E, (C^{(n)} = 0) 3 (C^{(n)} = 1) 2$
- 3:  $\bar{D} + 2 \rightarrow C, E \rightarrow D, 1 \rightarrow E, 4$
- 4:  $C + 1 \rightarrow C, ts(E) \rightarrow E, (C^{(n)} = 0) 5 (C^{(n)} = 1) 4$
- 5:  $E + \bar{D} + 1 \rightarrow D, A \rightarrow C, 0 \rightarrow A, (OR(A) = 0) 0$   
 $(A^{(n)}OR(A) = 01) 6 (A^{(n)}OR(A) = 11) 7$
- 6:  $\bar{C} + 1 \rightarrow C, 7$
- 7:  $D + A \rightarrow A, C + 1 \rightarrow C, (C^{(n)} = 0) 0 (C^{(n)} = 1) 7$
- 8:  $td(D + B) \rightarrow D, -1 \rightarrow B, 9$
- 9:  $td(D) \rightarrow D, B + 1 \rightarrow B, (OR(D) = 0) 10 (OR(D) = 1) 9$
- 10:  $C + 1 \rightarrow C, ts(B) \rightarrow B, (C^{(n)} = 0) 0 (C^{(n)} = 1) 10$

in cui si è realizzato come funzione logaritmo intero l'approssimazione per difetto del logaritmo reale, ovvero:

$$lg(n) = \lfloor \log_2(n) \rfloor$$

La parte relativa al codice operativo 0 poteva essere ottimizzata effettuando prima le moltiplicazioni di  $A$  per  $2^N$  e  $2^M$  e poi operando una sottrazione,

come illustrato nel microcodice seguente:

0 :  $A \rightarrow A, B \rightarrow B, \overline{N} + 2 \rightarrow C, M \rightarrow D, A \rightarrow E, 1$   
 1 :  $O_0, (K = 0) 2 (K = 1) 6$   
 2 :  $C + 1 \rightarrow C, tsar(E) \rightarrow E, (C^{(n)} = 0) 3 (C^{(n)} = 1) 2$   
 3 :  $\overline{D} + 2 \rightarrow C, E \rightarrow D, A \rightarrow E, 4$   
 4 :  $C + 1 \rightarrow C, tsar(E) \rightarrow E, (C^{(n)} = 0) 5 (C^{(n)} = 1) 4$   
 5 :  $D + \overline{E} + 1 \rightarrow A, 0$   
 6 : ...

#### Esercizio 4

Indichiamo con  $T$  la durata di un ciclo di clock ( $T = 1/f = 25\text{ns}$ ), con  $x$  il numero di cicli di clock in cui può essere completata l'operazione di lettura e con  $T_r$  il tempo di risposta dello slave. Abbiamo:

$$x \cdot T - (T_{ad} + T_{ds} + 1/2 \cdot T) \geq T_r \quad (1)$$

da cui, ricavando  $x$

$$x \geq \frac{T_{ad} + T_{ds} + 1/2 \cdot T + T_r}{T} = \frac{117.5}{25} = 4.7$$

poichè  $x$  deve essere intero abbiamo  $x = 5$ . Il numero di cicli di cui può essere accorciata l'operazione di lettura è pari a  $8-5=3$ . Per quanto riguarda il tempo massimo di risposta per uno slave occorre riutilizzare la (1) con  $x = 8$  e  $T_r$  come incognita. Otteniamo:

$$T_r \leq 8 \cdot 25 - (9 + 6 + 12.5) \text{ ns} = 172.5 \text{ ns}$$

#### Esercizio 5

Il programma opera sulle locazioni di memoria che vanno dall'indirizzo 100 all'indirizzo 499, interpretandole come un insieme di 100 dati ciascuno da 4 byte. Ciò che fa è scambiare il dato 1 con il 100, il 2 con il 99, il 3 col 98 e così via. Infatti la prima parte inserisce in pila queste 100 parole da 4 byte, mentre la seconda le estrae, mantenendo però lo stesso ordine. Per la struttura *Last In First Out* della pila, questo ha l'effetto di scambiare i valori come descritto sopra.