

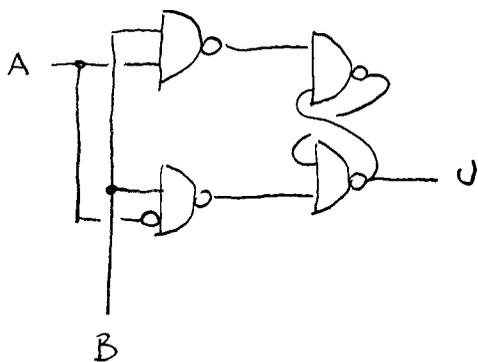
ESERCIZIO N°5

8 punti

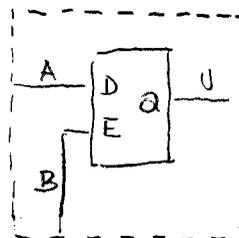
Una SRAM da 64k x 8 è collegata esternamente alle porte di un microcontrollore AT90S8515. In particolare, il bus degli indirizzi è collegato alle porte A (parte più significativa) e B (parte meno significativa), il bus dei dati alla porta C e i segnali di controllo $\overline{R/W}$ e \overline{CE} rispettivamente ai pin D0 e D1 della porta D. Scrivere la parte di inizializzazione del firmware che predisponga lo stack per il corretto funzionamento, e configuri le porte A, B e C in uscita con valore nullo, e i pin D0 e D1 in uscita con valore 1 (non attivo). Scrivere quindi un sottoprogramma in grado di leggere dalla memoria all'indirizzo contenuto nel puntatore X trasferendo il valore letto nel registro R16. Il sottoprogramma non deve alterare gli altri registri e deve lasciare l'interfaccia verso la memoria nello stato iniziale.

1

Si riconoscano nel circuito un inverter e 4 porte NAND.
 Si può ricostruire lo schema logico

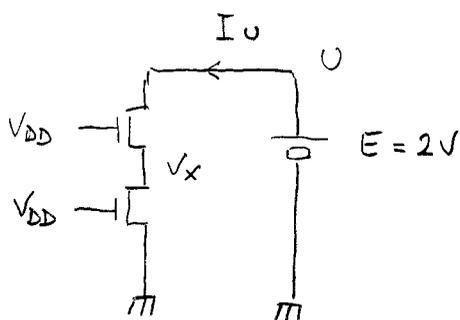


Si riconosce lo schema di un D-LATCH in cui A corrisponde a D e B corrisponde a E



Con ingresso $A=B=V_{DD}$ l'uscita viene posta a Φ . Applicando all'uscita un generatore si ha:

pMOS interdetti



nMOS in zona triodo perché entrambe le V_{GS} sono maggiori di V_{TM} .

$$\begin{cases} I_U = \frac{K_M}{2} (E - V_x)(2V_{DD} - V_x - E - 2V_{TM}) \\ I_U = \frac{K_M}{2} V_x (2V_{DD} - V_x - 2V_{TM}) \end{cases}$$

Uguagliando i due termini

$$(2 - V_x)(6 - V_x) = V_x(8 - V_x) \quad ; \quad 2V_x^2 - 16V_x + 12 = 0$$

$$V_x = 4 - \sqrt{16 - 6} = 0,8377 \quad (\text{la soluzione maggiore di } 2V \text{ non \u00e9 accettabile})$$

Quindi

$$I_U = (4 - \sqrt{10})(4 + \sqrt{10}) = 6 \text{ mA}$$

Note: si pu\u00f2 dimostrare che due MOS in serie con la stessa V_G si comportano come un unico MOS con

$$K_{eq} = K_1 K_2 / (K_1 + K_2)$$

2

Coverire esprimere direttamente la funzione in una mappa di Karnaugh

$x_2 = 0$

		$x_1 x_0$			
		00	01	11	10
$y_1 y_0$	1	0	1	1	0
	2	0	0	1	1
	4	0	0	0	0
	3	0	0	1	0

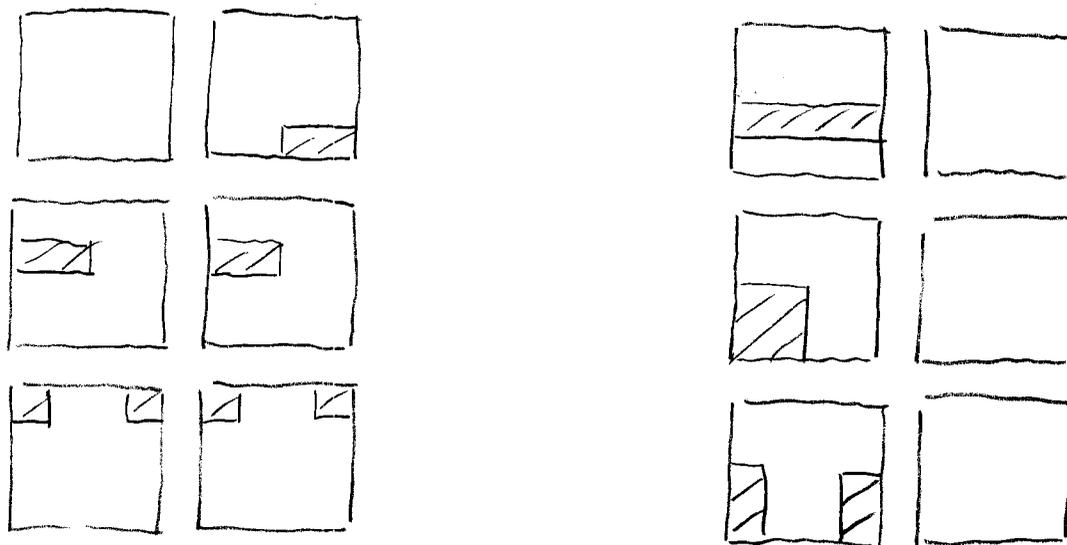
$\uparrow (Y+1)$
 $(X) \rightarrow$ 0 1 3 2

$x_2 = 1$

		$x_1 x_0$			
		00	01	11	10
$y_1 y_0$	00	0	1	1	0
	01	0	0	1	1
	11	1	1	1	1
	10	1	1	0	0

4 5 7 6

Identificazione della copertura ottima (ce ne sono diverse)



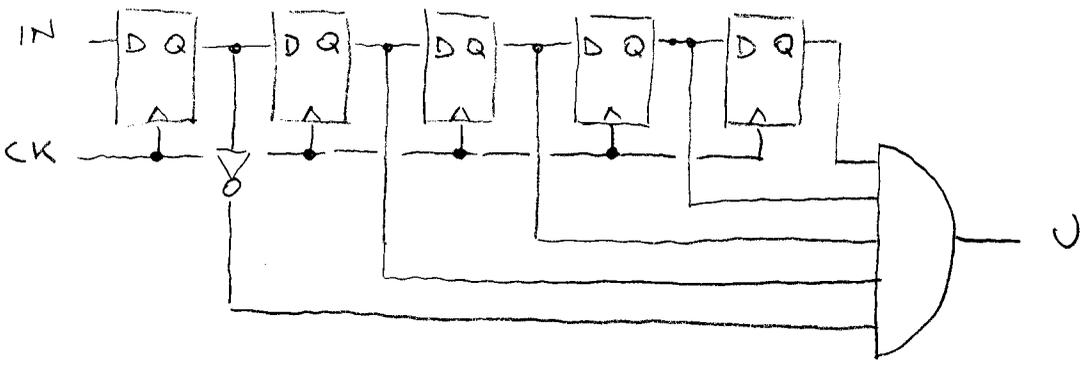
Forma PS

$$\begin{aligned}
 U = & (\bar{x}_2 + \bar{x}_1 + \bar{y}_1 + y_0) (x_1 + y_1 + \bar{y}_0) (x_0 + y_1 + y_0) \cdot \\
 & \cdot (x_2 + \bar{y}_1 + \bar{y}_0) (x_2 + x_1 + \bar{y}_1) (x_2 + x_0 + \bar{y}_1)
 \end{aligned}$$

3

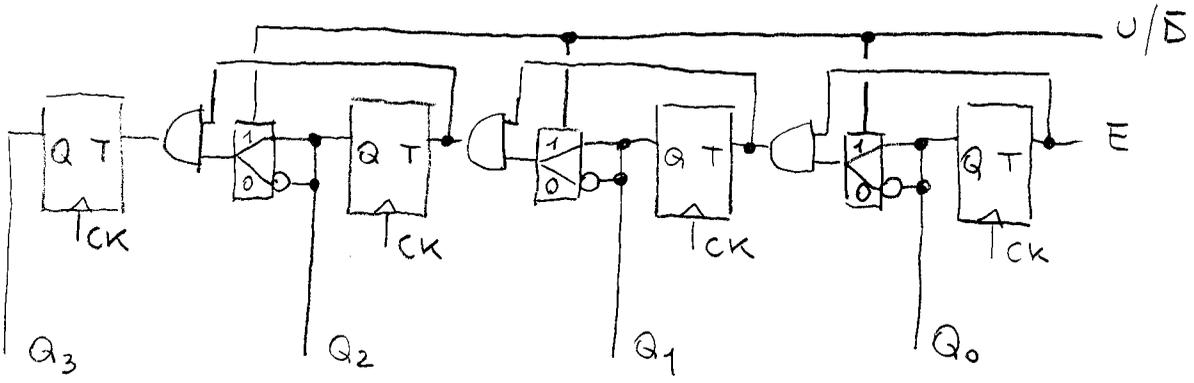
Il problema proposto è risolto da una macchina che riconosce la sequenza 11110. Non ci sono problemi di interacciamiento.

Sintesi ad hoc (di Moore, in quanto l'uscita dipende dallo stato)



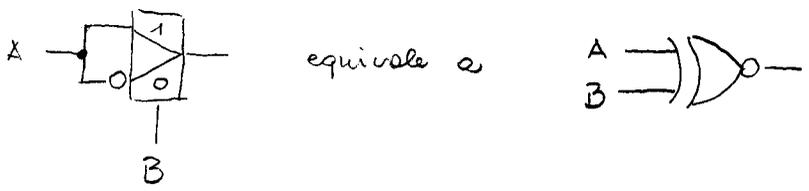
4

Contatore U/D con E a 4 bit. Uso multiplexer per selezionare l'ingresso opportuno per i flip-flop T.



- le flip-flop di ciascuna cifra commutano se (e solo se)
 - la cifra precedente è 1 (0 nel caso DOWN) e
 - il flip-flop precedente commuta

Nota: la rete logica



5

Inizializzazione

```
LDI R16, low(RAMEND) ; iniz. SP. vedi datasheet p.20
OUT SPL, R16
LDI R16, high(RAMEND)
OUT SPH, R16

SER R16 ; predisporre in uscita le porte
OUT DDRA, R16
OUT DDRB, R16
OUT DDRC, R16
SBI DDRD, 0
SBI DDRD, 1

CLR R16 ; predisporre i valori richiesti
OUT PORTA, R16
OUT PORTB, R16
OUT PORTC, R16
SBI PORTD, 0
SBI PORTD, 1

:
```

Sottoprogramma di lettura

```
read: PUSH R17
OUT PORTA, XH ; mette in uscita l'indirizzo
OUT PORTB, XL
CLR R17
OUT DDRC, R17 ; dati in ingresso
CBI PORTD, 1 ; accende la memoria
IN R16, PINC ; legge il dato
SBI PORTD, 1 ; spegne la memoria
SER R17
OUT DDRC, R17 ; rimette la porta dati nello stato iniz.
POP R17
RET
```