

Cognome

Nome

Matricola

ESERCIZIO N 1

7 punti (4)

Determinare, se possibile, una configurazione circuitale facente uso di un resistore di valore opportuno tale da rendere possibile il pilotaggio di una porta della famiglia logica B da parte di una porta della famiglia logica A , con $MNH = NML$. Individuare l'intervallo dei valori accettabili per la resistenza. In caso di risposta positiva al primo quesito, individuare il valore della resistenza che rende massimo il fan-out e il valore del fan-out stesso in questo caso.

$$V_{CC} = 5 \text{ V}; V_{OLA} = 0,6 \text{ V}; V_{OHA} = 3 \text{ V}; V_{ILB} = 1,2 \text{ V}; V_{IHB} = 4 \text{ V}; I_{OLA} = 20 \text{ mA}; I_{OHA} = -5 \text{ mA}; \\ I_{ILB} = -0,1 \text{ mA}; I_{IHB} = 0,1 \text{ mA}.$$

ESERCIZIO N 2

6 punti (4)

Realizzare in forma SP ottima una rete combinatoria a 5 ingressi (x_2, x_1 e x_0 e y_1, y_0) e una uscita che indica con 0 tutte le situazioni in cui esiste un k intero non negativo per cui $X = kY$. X e Y sono gli interi assoluti che corrispondono ai bit di nome analogo.

ESERCIZIO N 3

6 punti (4)

Realizzare una macchina sequenziale sincrona secondo il modello di Moore con 2 ingressi (IN e R) e una uscita Q in grado di riconoscere tre valori consecutivi diversi tra loro di IN (in corrispondenza di 3 fronti in salita di clock successivi) ponendo e mantenendo a 1 l'uscita Q , che viene comunque azzerata, senza tenere alcun conto del valore contestuale di IN , dalla presenza di un 1 in ingresso a R .

ESERCIZIO N4

6 punti (4)

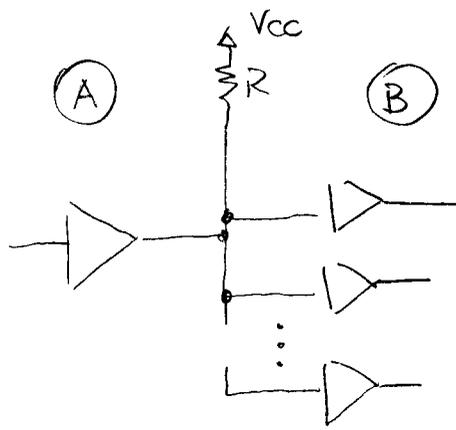
Assemblare memorie SRAM da $1M \times 2$ e $512k \times 5$ in modo da ottenere una memoria da $1M \times 16$. Si individui il minimo numero di chip possibile per ottenere il risultato.

ESERCIZIO N5

8 punti (4)

Scrivere un sottoprogramma nel linguaggio assembly della famiglia XMEGA AVR che determina quante volte un carattere, il cui codice è dato nel registro $R16$, è contenuto nella stringa di N caratteri consecutivi (il valore di N , che può essere anche 0, è dato in $R17$) posto nella memoria dati estesa a partire dall'indirizzo contenuto in Z . Il risultato va lasciato nello stesso $R17$. Il sottoprogramma non deve alterare i registri di lavoro.

①



Non è rispettata la condizione sul livello logico alto

$$V_{OHA} > V_{IHB}$$

Quindi proviamo a porre un resistore di pull-up.
In queste condizioni dovrà essere

$$\begin{cases} V_{CC} - RNI_{IHB} \geq V_{IHB} + NML & (NML = V_{ILB} - V_{OLA} = 0,6V) \\ I_{OLA} \geq N|I_{ILB}| + \frac{V_{CC} - V_{OLA}}{R} \end{cases}$$

$$\frac{V_{CC} - V_{OLA}}{I_{OLA} - N|I_{ILB}|} \leq R \leq \frac{V_{CC} - V_{IHB} - NML}{NI_{IHB}}$$

Sostituisco i valori numerici

$$\frac{4,4}{20 - 0,1N} \leq R (k\Omega) \leq \frac{0,4}{0,1N}$$

$$\frac{44}{200 - N} \leq R (k\Omega) \leq \frac{4}{N}$$

per $N=1$ $0,219 k\Omega \leq R \leq 4 k\Omega$

massimo valore di N per cui si ha soluzione

$$\frac{N_{MAX}}{4} \leq \frac{200}{44} - \frac{N_{MAX}}{4,4} \quad ; \quad N_{MAX} \leq \frac{200}{12} \quad ;$$

$$N_{MAX} = 16$$

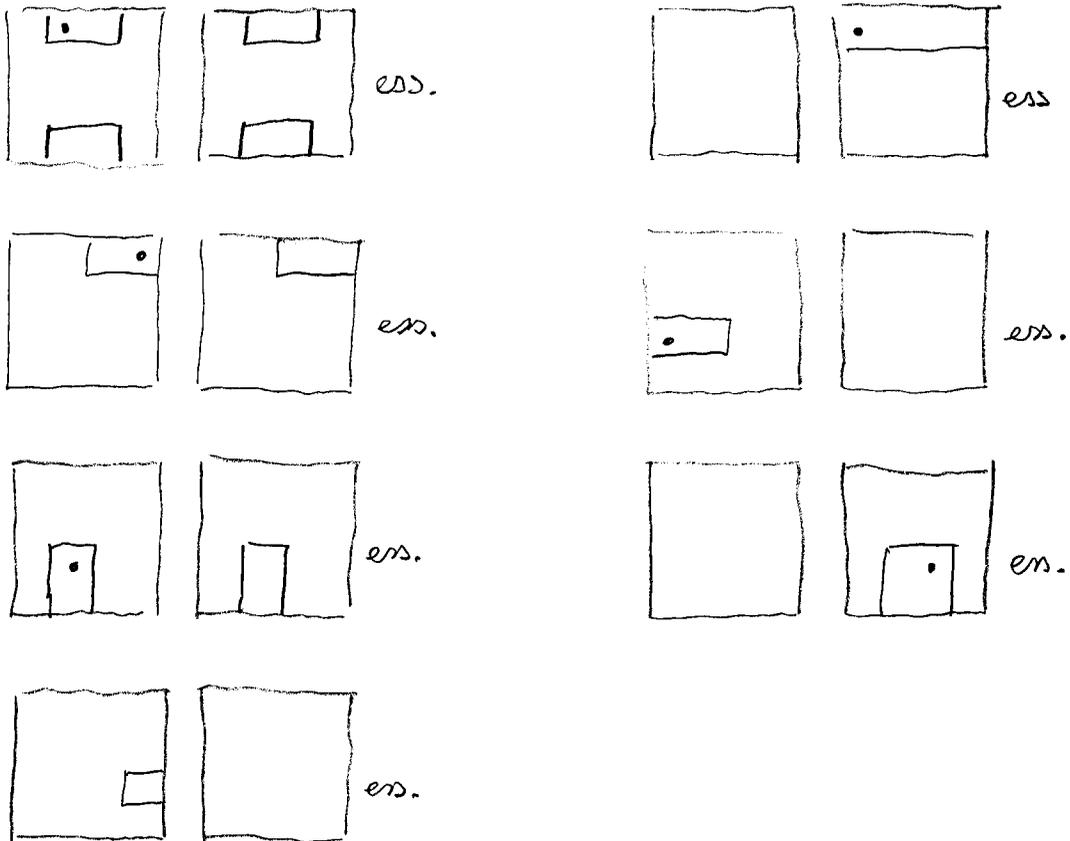
②

$$U = \overline{(x = ky)} \text{ con } k \in \mathbb{N}$$

		$x_2 = 0$			
	$x_1 x_0$	00	01	11	10
Y	$Y_1 Y_0$	00	01	11	10
	00	0	1	1	1
	01	0	0	0	0
	11	0	1	0	1
	10	0	1	1	0
		0	1	3	2

		$x_2 = 1$			
	$x_1 x_0$	00	01	11	10
Y	$Y_1 Y_0$	00	01	11	10
	00	1	1	1	1
	01	0	0	0	0
	11	1	1	1	0
	10	0	1	1	0
		4	5	7	6

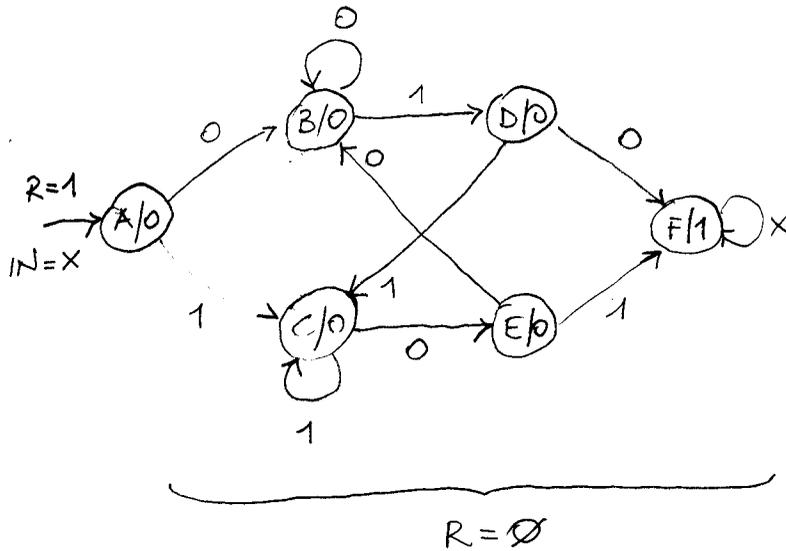
Stiles: SP



$$U = x_0 \bar{y}_0 + x_1 \bar{y}_1 \bar{y}_0 + \bar{x}_1 x_0 y_1 + \bar{x}_2 x_1 \bar{x}_0 y_1 y_0 + x_2 \bar{y}_1 \bar{y}_0 + x_2 \bar{x}_1 y_1 y_0 + x_2 x_0 y_1$$

3

grafo

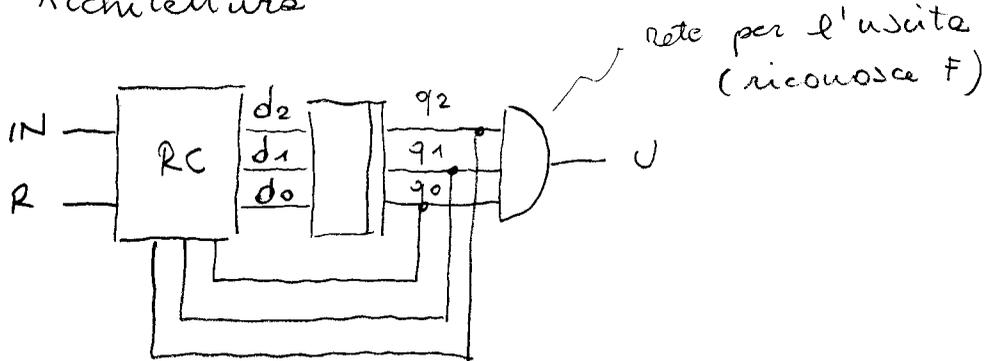


Codifica

A	000
B	100
C	001
D	110
E	011
F	111

G	010
H	101

Architettura



q_2, IN

q_2, IN	00	01	11	10
00	100	001	110	100
01	011	001	110	110
11	001	111	111	111
10	100	001	001	111

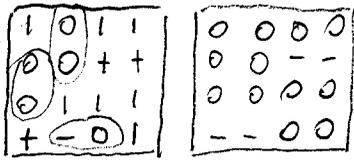
$R=0$

q_2, IN

	00	01	11	10
00	000	000	000	000
01	000	000	000	000
11	000	000	000	000
10	000	000	000	000

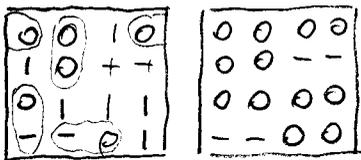
$R=1$

Sintesi d_2 (PS)



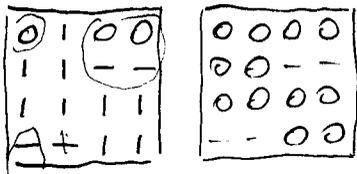
$$d_2 = \bar{R} (\bar{1N} + q_2 + q_1) (1N + q_2 + \bar{q}_0) (\bar{1N} + \bar{q}_1 + q_0)$$

Sintesi d_1 (PS)



$$d_1 = \bar{R} (1N + q_1 + q_0) (\bar{1N} + q_2 + q_1) \cdot (1N + q_2 + \bar{q}_1) (\bar{1N} + \bar{q}_1 + q_0)$$

Sintesi d_0 (PS)

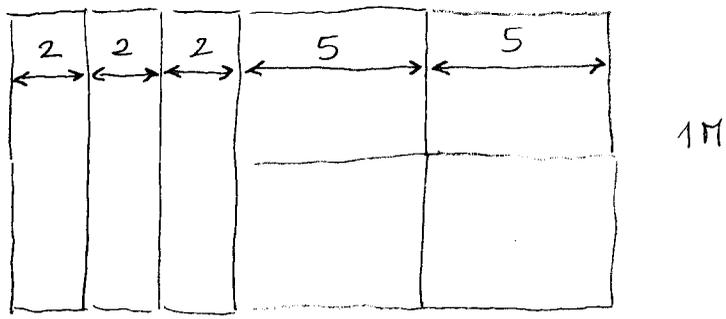


$$d_0 = \bar{R} (\bar{q}_2 + q_1) (1N + q_2 + q_0)$$

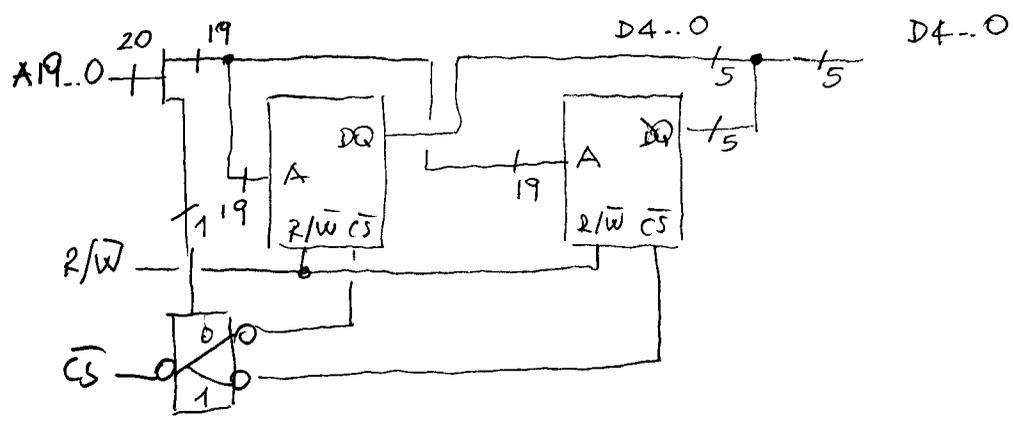
A posteriori possiamo osservare che gli stati H e G non danno problemi di blocco.

4

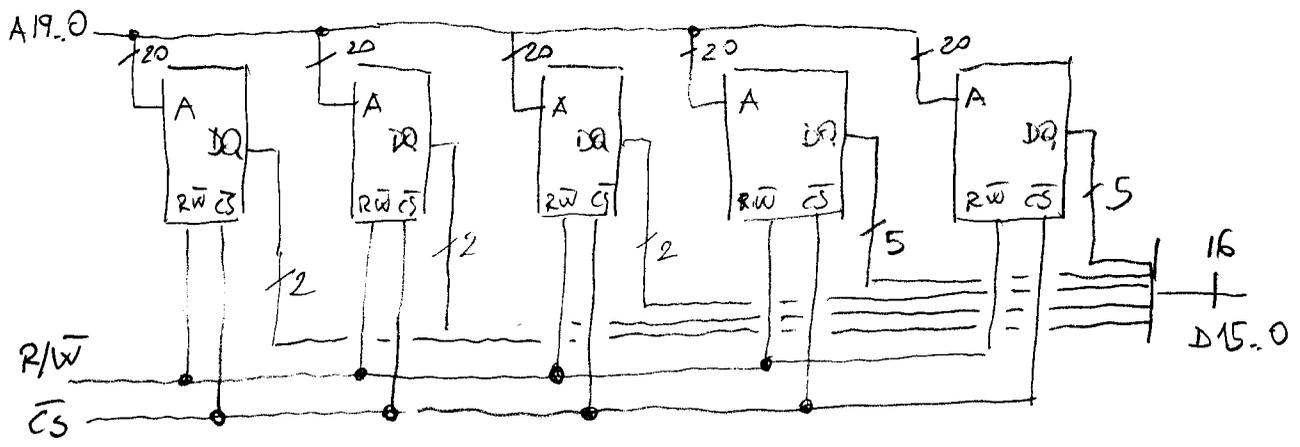
Occorrono 7 chip al minimo.
Soluzione senza sprechi



Aumento n° parole 2 da 512K x 5 → 1M x 5



Aumento dimensione parole



5

```
/* Cerca un particolare carattere (in R16) nella stringa puntata da X, di lunghezza N (R17).
   Risultato in R17.
*/
ricerca:
    tst R17                //controlla la lunghezza di stringa
    breq fine             //se 0, va bene uscire direttamente
    push R18              //salva il contatore
    push R19              //salva registro di appoggio
    push XL
    push XH
    mov R18,R17
    clr R17               //predispone il risultato
loop:
    ld R19,X+            //carica il primo carattere
    cp R16,R19           //esegue il confronto
    brne poi
        inc R17          //incrementa se il carattere è quello cercato
    poi:
        dec R18
    brne loop
    pop XH
    pop XL
    pop R19
    pop R18
    fine:
ret
```