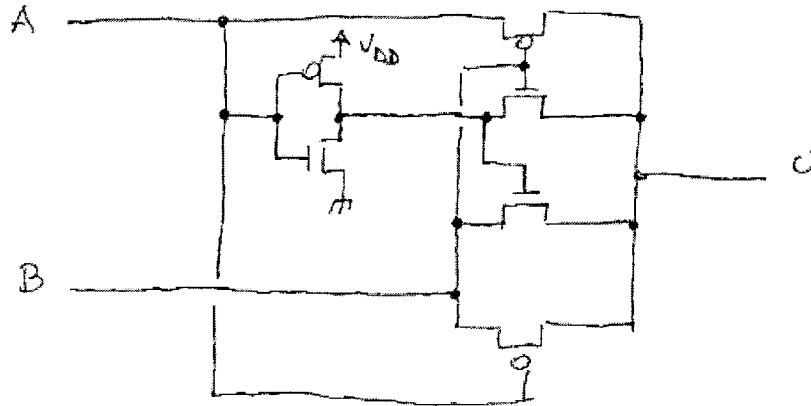


ESERCIZIO N°1

7 punti (3)

Determinare la funzione logica della seguente rete CMOS e la corrente erogata a regime dall'ingresso A collegato all'alimentazione ($V_{DD} = 5\text{ V}$) quando l'altro ingresso e l'uscita sono collegati a massa. Si ha $V_{Tn} = 1\text{ V}$; $V_{Tp} = -1\text{ V}$; $k_n = 4\text{ mA/V}^2$; $k_p = -4\text{ mA/V}^2$.



ESERCIZIO N°2

6 punti (4)

Realizzare, possibilmente con un approccio modulare, una rete combinatoria a 6 ingressi (2 parole di 3 bit) in grado di indicare con un valore alto in uscita quando uno dei due ingressi è maggiore o uguale all'altro.

ESERCIZIO N°3

6 punti (3)

Collegare delle memorie da 1G x 3 (costo 3€) e 512Mx 2 (costo 1,2€) in modo da ottenere un modulo da 1G x 8 di costo minimo.

ESERCIZIO N°4

6 punti (4)

Disegnare lo schema logico di un sequenziatore con contatore sincrono (dotato della possibilità di caricamento parallelo) che implementi microcodice specificato nel seguito.

```

A:   IF J THEN C ELSE F; OP = 101
B:   IF K THEN F ELSE G; OP = 100
C:   IF M THEN D ELSE H; OP = 010
D:   IF M THEN E ELSE C; OP = 111
E:   IF L THEN H ELSE D; OP = 110
F:   IF L THEN A ELSE B; OP = 000
G:   IF J THEN B ELSE A; OP = 011
H:   IF K THEN G ELSE G; OP = 001
    
```

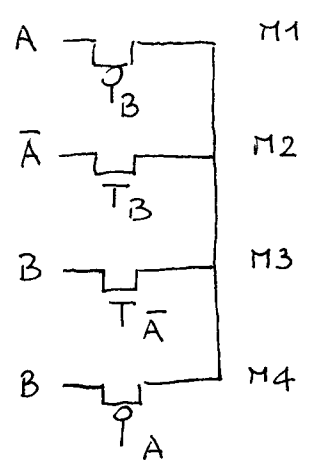
ESERCIZIO N°5

8 punti (4)

Realizzare un sottoprogramma assembly per un microcontrollore della famiglia AVR che cambia il segno al contenuto di X, inteso come valore in complemento a 2 su 16 bit. Individuare in che caso si ha la condizione di overflow e fare in modo che, al ritorno dalla subroutine, questa eventualità sia riportata da un valore vero del flag V.

1

Funzione logica.
 La situazione è la seguente

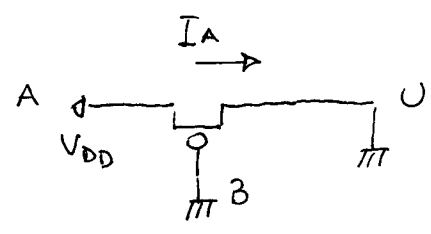


A	B	M1	M2	M3	M4	U
0	0	on ₀	off	on ₀	on ₀	0
0	1	off	on ₁	on ₁	on ₁	1
1	0	on ₁	off	off	off	1
1	1	off	on ₀	off	off	0

La funzione è una XOR

(il pedice indica il valore che viene trasmesso)

Situazione per la corrente: è acceso solo M1 (e il PMOS della NOT, irrilevante)

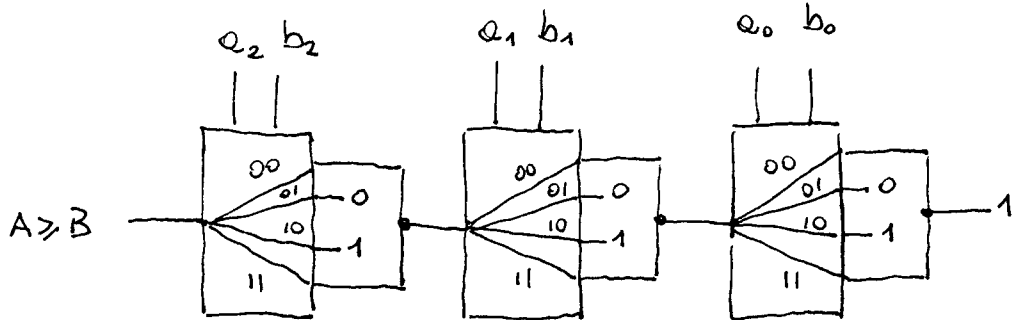


$V_{GS} = -V_{DD}$
 $V_{GD} = \phi$ Saturato

$$I_A = -\frac{K_P}{2} (V_{DD} + V_{TP})^2 = 32 \text{ mA}$$

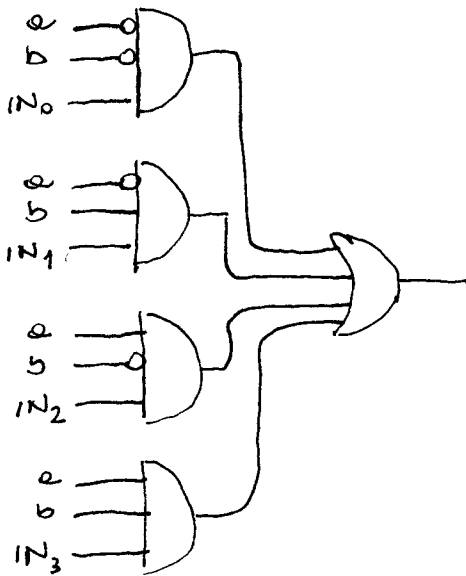
2

Una possibile soluzione MODULARE è la seguente



A ogni bit, se c'è uguaglianza il mux propaga il valore precedente, altrimenti decide quale è la disuguaglianza fino a quel punto.

MUX a porte



(Vedere i comparatori; Ciroue p. 246)

③ Per volutare l'assemblaggio più conveniente, si possono riportare i due tipi di chip a un modulo con lo stesso numero di parole

a) $1G \times 3$ costo $1 \text{€}/Gb$ unitario

b) $2(512M \times 2) \rightarrow 1G \times 2$ costo $1,2 \text{€}/Gb$

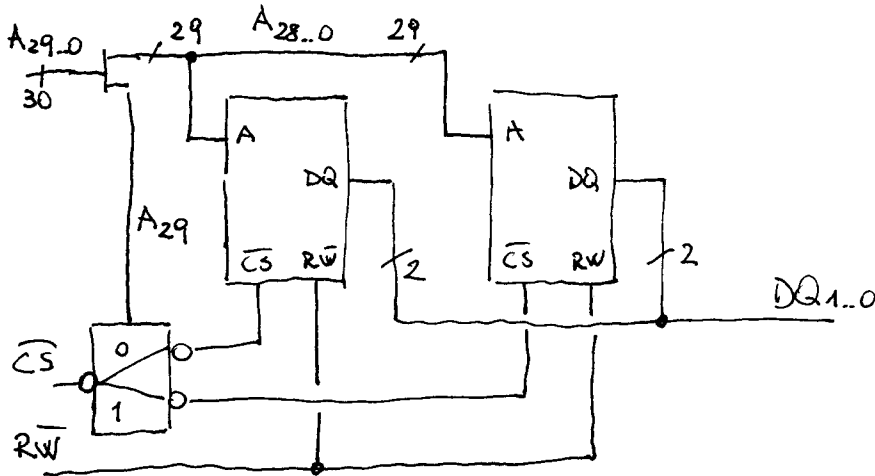
Quindi il tipo a ha un costo per bit minore, pertanto conviene scegliere le configurazioni che lo privilegiano

Per ottenere $1G \times 8$ si possono usare

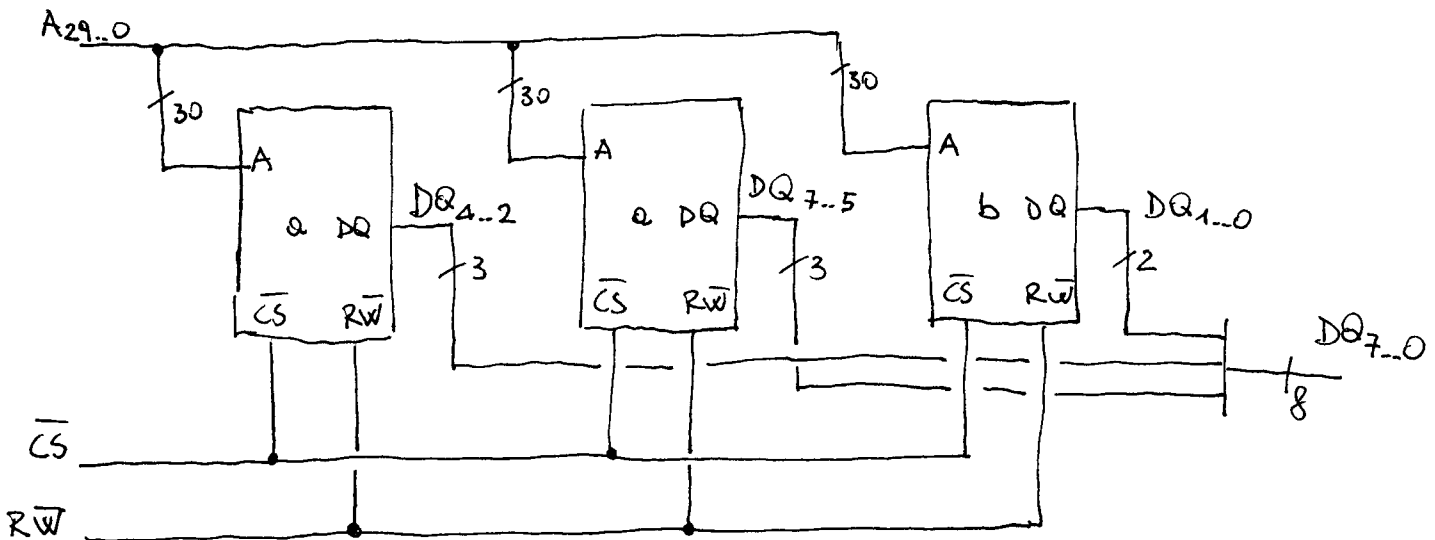
3 a (con spreco di $1Gb$) $\rightarrow 9 \text{€}$

2 a + b (senza spreco) $\rightarrow 8,4 \text{€}$ soluzione OTTIMA

Montaggi



Aumento n° parole da
512M a 1G



④

Sequenza ciclica completa (se vero)

A, C, D, E, H, G, B, F, ...

Codifico gli stati in questo ordine e riorganizzo la descrizione

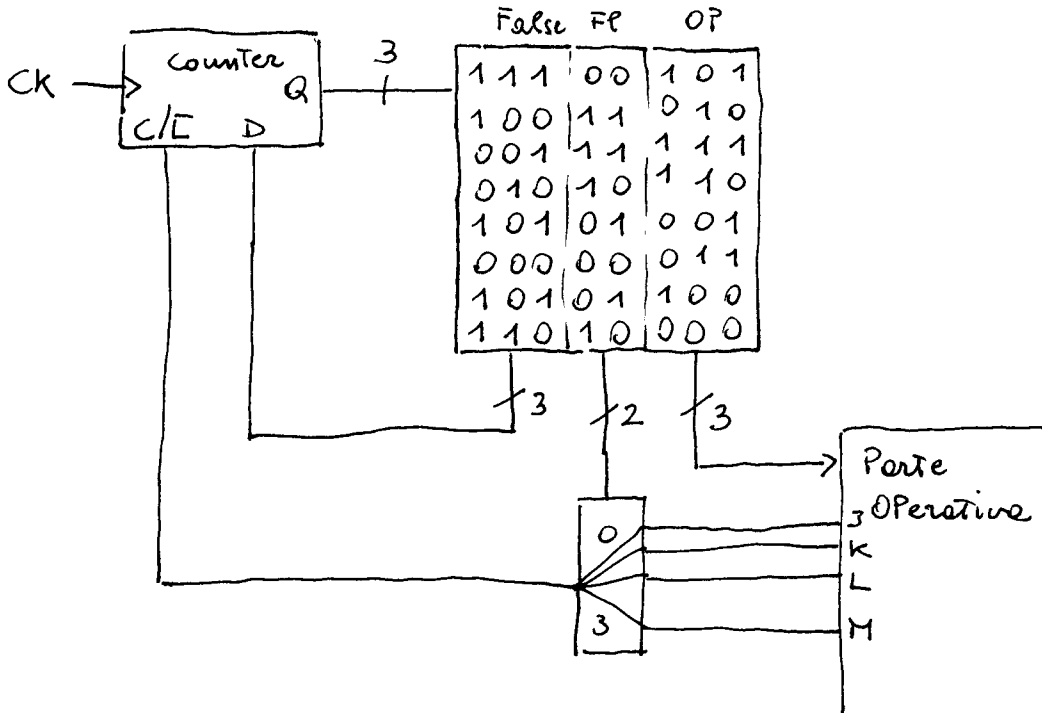
A	000	F	J	101
C	001	H	M	010
D	010	C	M	111
E	011	D	L	110
H	100	G	K	001
G	101	A	J	011
B	110	G	K	100
F	111	B	L	000

codif. flag

J	00
K	01
L	10
M	11

x F Flag OPCODE

Schema logico



5

```
chs: NEG XL  
      BRCS skip  
      NEG XH  
      RJMP end
```

```
skip: COM XH  
end:  RET
```

// non c'è carry solo per $XL = \phi$
// sistema anche \checkmark per -2^{15}

// se $XL \neq \phi$ va usata la COM