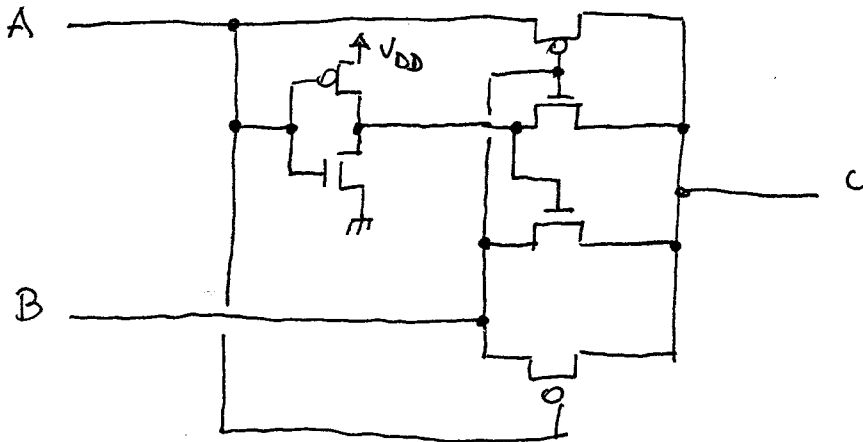


### ESERCIZIO N°1

7 punti (4)

Individuare il funzionamento del seguente circuito logico CMOS. Valutare quindi la corrente di uscita nel caso in cui venga posto in uscita un generatore ideale di tensione da 2 V e gli ingressi A e B valgano entrambi 5V. ( $V_{DD} = 5\text{ V}$ ;  $V_{Tn} = |V_{Tp}| = 1\text{ V}$ ;  $k_n = |k_p| = 2\text{ mA/V}^2$ ).



### ESERCIZIO N°2

5 punti (5)

Realizzare in forma PS ottima una rete combinatoria a 4 ingressi ( $x_1, x_0$  e  $y_1, y_0$ ) e una uscita che indica con 1 se  $X \geq Y$ . X e Y sono interpretati come numeri interi assoluti composti dai corrispondenti bit.

### ESERCIZIO N°3

7 punti (4)

Realizzare una macchina sequenziale sincrona secondo il modello di Moore con 1 ingresso e 1 uscita viene posta a 1 (e poi resta tale) al termine di ogni impulso del segnale di ingresso di durata superiore a 2 cicli di clock e torna a 0 (e poi resta tale) se l'ingresso si mantiene 0 per più di 2 cicli di clock, anche in questo caso al termine dell'impulso negativo.

### ESERCIZIO N°4

6 punti (3)

Realizzare, usando un registro a 3 bit e una memoria RAM opportuna, un sequenziatore in grado di generare una sequenza di codici operativi assimilabile a quello prodotta da un contatore UP/!DOWN modulo 8. Il segnale di controllo è un flag proveniente dalla parte operativa.

### ESERCIZIO N°5

8 punti (5)

Tre pulsanti T1, T2 e T3 hanno i due terminali collegati T1 ai pin B4 e B7, T2 ai pin B5 e B7, e T3 ai pin B6 e B7 della porta B di un microcontrollore AT90S8515. Inoltre tre LED, L1, L2 e L3, hanno l'anodo collegato all'alimentazione e i catodi, in serie a una opportuna resistenza, ai pin della stessa porta B0, B1 e B2 rispettivamente. Scrivere un programma che, dopo avere inizializzato correttamente le porte, indichi (continuamente) con un LED acceso se il corrispondente pulsante è premuto.

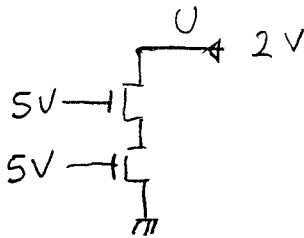
9

Funzione logica: esaminiamo i veri casi

A	B	U	P1	N1	N2	P2
0	0	0	on	off	on	on
0	1	1	off	on	on	on
1	0	1	on	off	off	off
1	1	0	off	on	off	off

$$U = A \oplus B$$

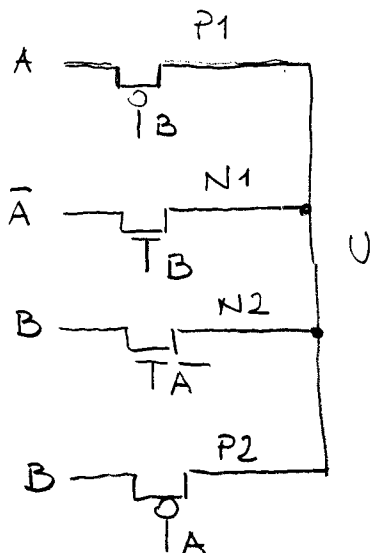
Circuito con  $A = B = 5V$ .



$$K_{eq} = K_m / 2 = 1 \text{ mA/V}^2$$

Siamo in zona triodo; la corrente è entrante e vola

$$I_{DS} = \frac{K_{eq}}{2} \cdot V_U (V_{DD} + V_{DD} - V_U - 2V_{TM}) = 6 \text{ mA}$$



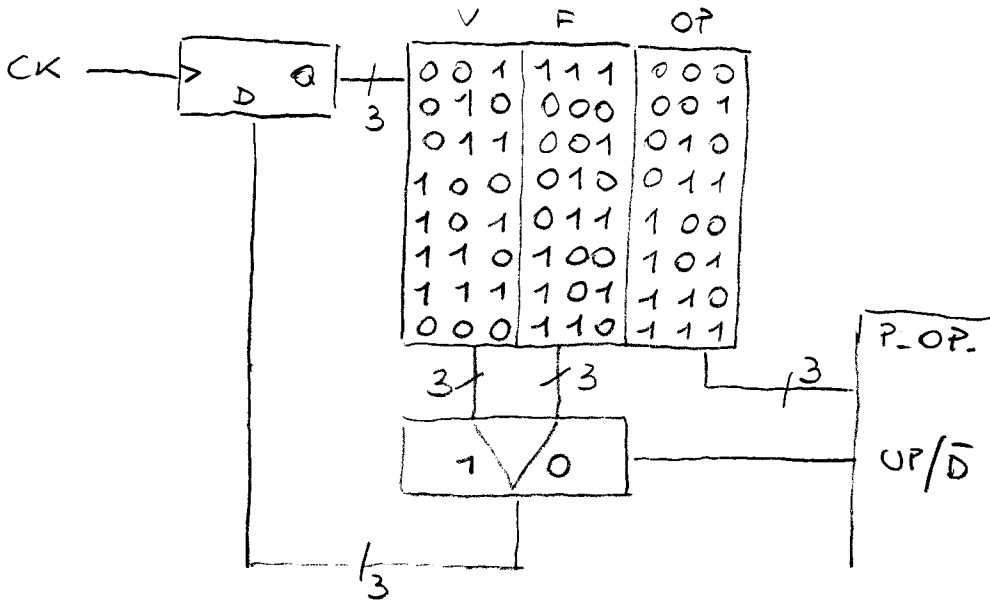
②

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

$F = X \geq Y$

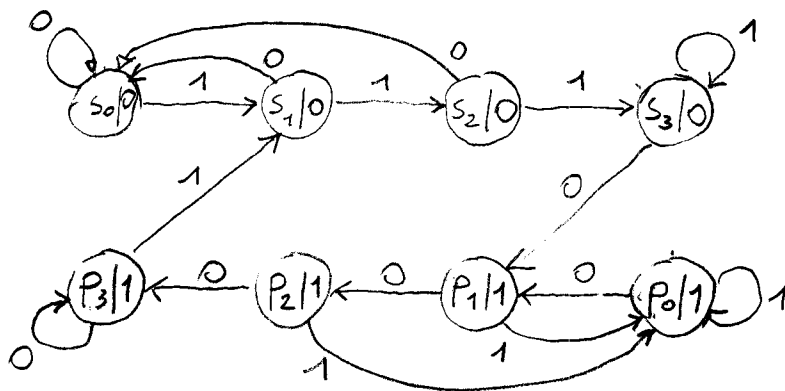
$F = (x_1 + \bar{y}_1)(x_1 + x_0 + \bar{y}_0)(\bar{y}_1 + \bar{y}_0 + x_0)$

④



Il codice operativo coincide con lo stato Q ; potrebbe essere eliminato dalla ROM ;

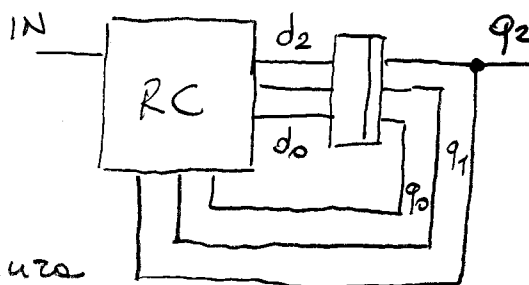
③ grafo (componente simmetrico)



codifica degli stati (q<sub>2</sub> coincide con l'uscita)

	q <sub>2</sub>	q <sub>1</sub>	q <sub>0</sub>
S <sub>0</sub>	0	0	0
S <sub>1</sub>	0	0	1
S <sub>2</sub>	0	1	0
S <sub>3</sub>	0	1	1

	q <sub>2</sub>	q <sub>1</sub>	q <sub>0</sub>
P <sub>0</sub>	1	0	0
P <sub>1</sub>	1	0	1
P <sub>2</sub>	1	1	0
P <sub>3</sub>	1	1	1



Architettura

Sintesi di RC

q <sub>1</sub> q <sub>0</sub> \ IN, q <sub>2</sub>	00	01	11	10
00	000	101	100	001
01	000	110	100	010
11	101	111	001	011
10	000	111	100	011

d <sub>2</sub>	d <sub>1</sub>	
0 1 1 0	0 0 0 0	0 1 0 1
0 1 1 0	0 1 0 1	0 0 0 0
1 1 0 0	0 1 0 1	1 1 1 1
0 1 1 0	0 1 0 1	0 1 0 1

$$d_2 = q_2 \bar{q}_1 + q_2 \bar{q}_0 + \bar{IN} q_1 q_0$$

$$d_1 = \bar{IN} q_2 q_0 + \bar{IN} q_2 q_1 + IN \bar{q}_2 q_0 + IN \bar{q}_2 q_1$$

$$d_0 = q_1 q_0 + \bar{IN} q_2 \bar{q}_0 + IN \bar{q}_2 \bar{q}_0$$

5

```
; Tested with VMLAB
; Program starts here after Reset
;
start:
    ldi R16, 0b10000111    ; inizializzo la direzione della porta B
    out DDRB, R16         ; in uscita led e pin a comune ai tasti
    ldi R16, 0b01111111   ; metto a GND il comune dei tasti, LED spenti
    out PORTB, R16
    ldi R17, 0b01111000   ; predispongo una maschera per la porta B

forever:
    in R16, PINB          ; tasti in ingresso (0 se premuti)
    andi R16, 0b01110000  ; isolo i bit con i tasti
    swap R16              ; li porto in corrispondenza con i LED
    add R16, R17          ; aggiungo i bit per pull-up e B7
    out PORTB, R16
    rjmp forever         ; loop infinito
```