

Cognome

Nome

Matricola

**ESERCIZIO N°1**

7/6 punti

Individuare nel piano cartesiano quotato (0,5 V/quadretto) della caratteristica di trasferimento le diverse regioni di funzionamento dei transistori NMOS e PMOS di un invertitore CMOS. ( $V_{DD} = 6$  V;  $V_{Th} = 1,5$  V;  $|V_{Tp}| = 1$  V;  $k_n = 8$  mA/V<sup>2</sup>;  $|k_p| = 4$  mA/V<sup>2</sup>)

**ESERCIZIO N°2**

6/4 punti

Realizzare in forma NAND-NAND ottima una rete combinatoria non completamente specificata a 4 ingressi e 1 uscita, i cui mintermini siano {0, 1, 4, 7, 8, 12, 15} e l'insieme dei don't care sia {2, 3, 5, 9}.

**ESERCIZIO N°3**

5/5 punti

Disegnare lo schema di un registro universale a 4 bit, in grado di garantire caricamento parallelo, shift destro e sinistro e azzeramento.

**ESERCIZIO N°4**

7/4 punti

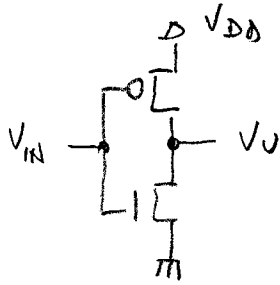
Progettare una rete di Moore in grado di generare la sequenza 101101. La macchina ha un ingresso di reset sincrono attivo alto.

**ESERCIZIO N°5**

8/5 punti

Realizzare una subroutine per un microcontrollore della famiglia AVR che traspone la matrice di byte di dimensioni  $3 \times 3$ , collocata in memoria dati estesa a partire dall'indirizzo contenuto in Z. La matrice trasposta sostituisce la matrice originale. La matrice è memorizzata riga per riga.

1



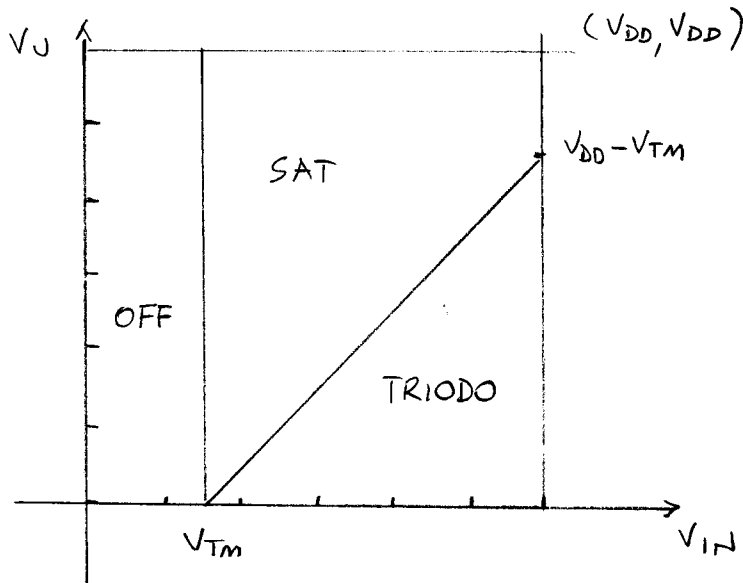
$$V_{GS} = V_{IN}$$

$$V_{GD} = V_{IN} - V_U$$

$$V_{GS} = V_{IN} - V_{DD}$$

$$V_{GD} = V_{IN} - V_U$$

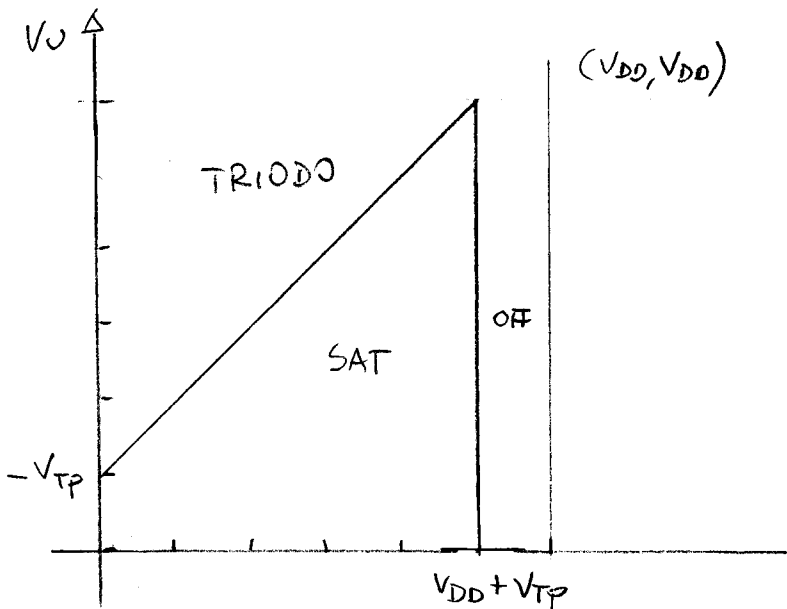
Regioni di funz. dell' NMOS



$$\text{OFF} : V_{IN} < V_{TM}$$

$$\text{SAT} : \begin{cases} V_{IN} > V_{TM} \\ V_{IN} - V_U < V_{TM} \end{cases}$$

Regioni del PMOS



$$\text{OFF} : V_{IN} - V_{DD} > V_{TP}$$

$$\text{SAT} : \begin{cases} V_{IN} - V_{DD} < V_{TP} \\ V_{IN} - V_U > V_{TP} \end{cases}$$

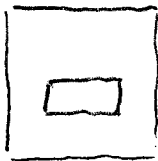
nota: le rette al confine tra SAT e TRIODO hanno pendenza UNITARIA (45°)

②

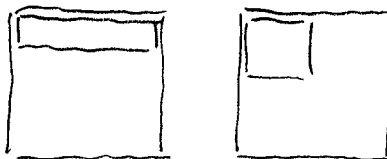
Forma NAND-NAND si ottiene applicando il teorema di De Morgan e una forma SP

$x_3x_2$ $x_1x_0$	00	01	11	10
00	1 <sup>0</sup>	1 <sup>4</sup>	1 <sup>12</sup>	1 <sup>8</sup>
01	1 <sup>1</sup>	- <sup>5</sup>	0 <sup>13</sup>	- <sup>9</sup>
11	- <sup>3</sup>	1 <sup>7</sup>	1 <sup>15</sup>	0 <sup>11</sup>
10	- <sup>2</sup>	0 <sup>6</sup>	0 <sup>14</sup>	0 <sup>10</sup>

Implicanti principali essenziali



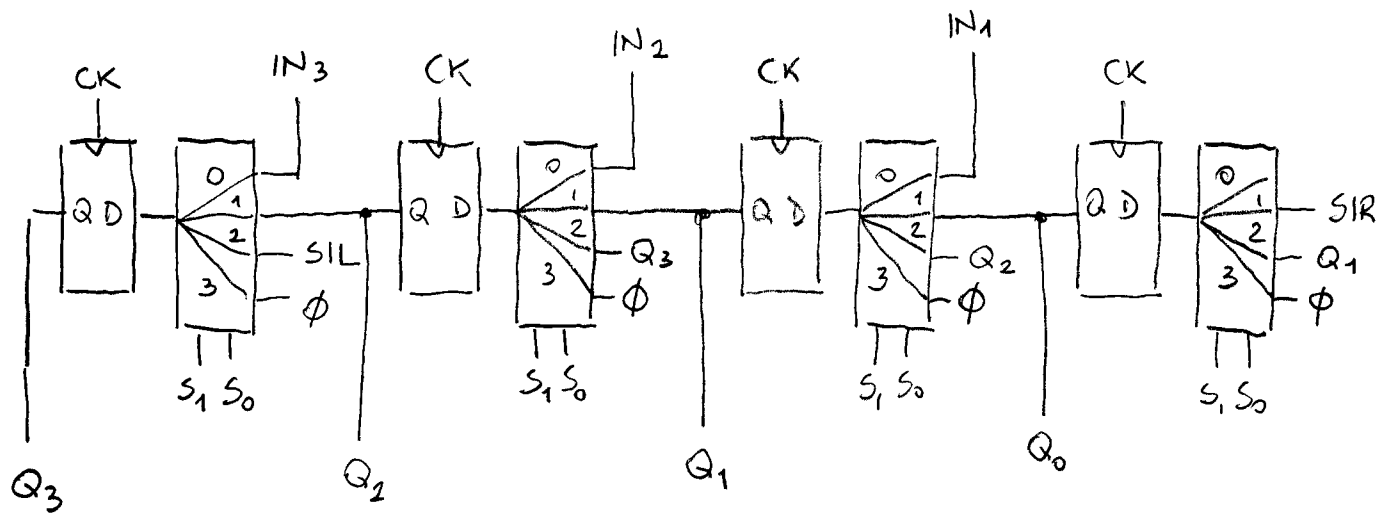
Per gli altri 1 bastano (e servono) 2 implicanti di ordine 2, per esempio



Quindi

$$U = \overline{(x_2x_1x_0 \cdot \overline{x_1}\overline{x_0} \cdot \overline{x_3}\overline{x_1})}$$

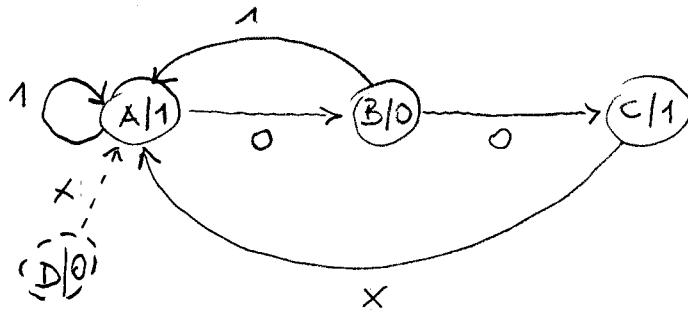
3



$S_1$	$S_0$	funzione
0	0	ingresso parallelo
0	1	shift sinistro (SIR: serial input Right)
1	0	shift destro (SIL: serial input Left)
1	1	reset

④

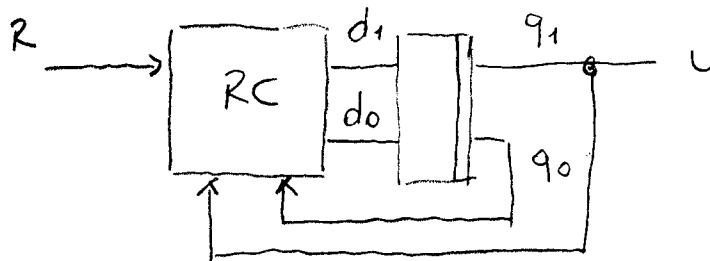
Grafo del GENERATORE (la seconda parte della sequenza è uguale alla prima) 101



Codifica

A	11
B	01
C	10
	q <sub>1</sub> q <sub>0</sub>

Con la codifica scelta q<sub>1</sub> coincide con l'uscita Architecture



Rete per lo stato futuro

R \ q <sub>1</sub> q <sub>0</sub>	00	01	11	10
0	--	10	01	11
1	--	11	11	11

$$d_1 = R + \bar{q}_1 + \bar{q}_0$$

$$d_0 = R + q_1 + \bar{q}_0$$

Note: se per un motivo qualsiasi si dovesse avere lo stato D (0,0), dopo il clock si avrebbe comunque A (1,1)

# 5

Transpose:

```
PUSH R16
PUSH R17
LDD R16,Z+1           //scambia a12 con a21
LDD R17,Z+3
STD Z+3,R16
STD Z+1,R17
LDD R16,Z+2           //scambia a13 con a31
LDD R17,Z+6
STD Z+6,R16
STD Z+2,R17
LDD R16,Z+5           //scambia a23 con a32
LDD R17,Z+7
STD Z+7,R16
STD Z+5,R17
POP R17
POP R16
RET
```