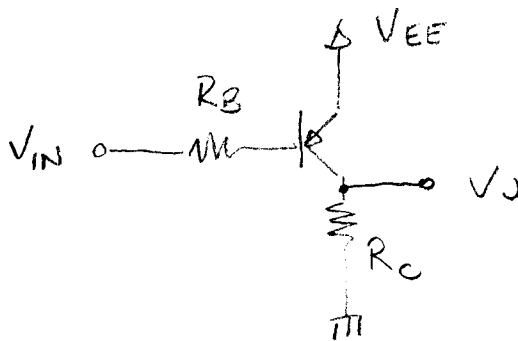


**ESERCIZIO N°1**

7/6 punti

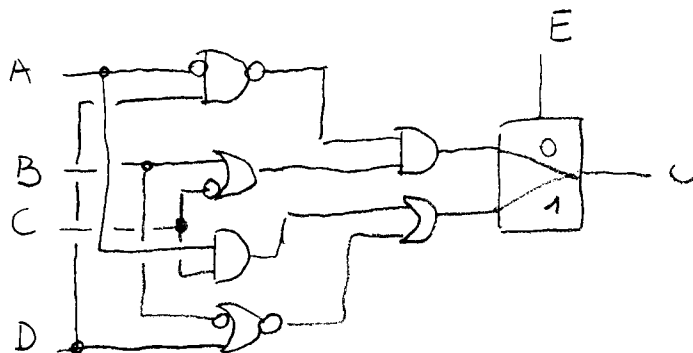
Determinare il fan-out con eguali margini di rumore del seguente invertitore RTL con transistor pnp. ( $h_{FE} = 200$ ;  $R_B = 15 \text{ k}\Omega$ ;  $R_C = 1,2 \text{ k}\Omega$ ;  $V_{EE} = 7 \text{ V}$ )



**ESERCIZIO N°2**

6/4 punti

Realizzare in forma NAND-NAND ottima la seguente rete combinatoria:



**ESERCIZIO N°3**

6/4 punti

Disegnare lo schema logico di un sequenziatore con contatore sincrono (dotato della possibilità di caricamento parallelo) che implementi microcodice specificato nel seguito. Si usi una ROM delle minime dimensioni possibili.

- A: IF J THEN D ELSE F; OP = 101111
- B: IF L THEN H ELSE E; OP = 100011
- C: IF K THEN F ELSE G; OP = 101101
- D: IF M THEN E ELSE H; OP = 010100
- E: IF M THEN B ELSE D; OP = 111101
- F: IF L THEN A ELSE C; OP = 011100
- G: IF J THEN C ELSE A; OP = 011100
- H: IF K THEN G ELSE G; OP = 010010

## **ESERCIZIO N°4**

6/4 punti

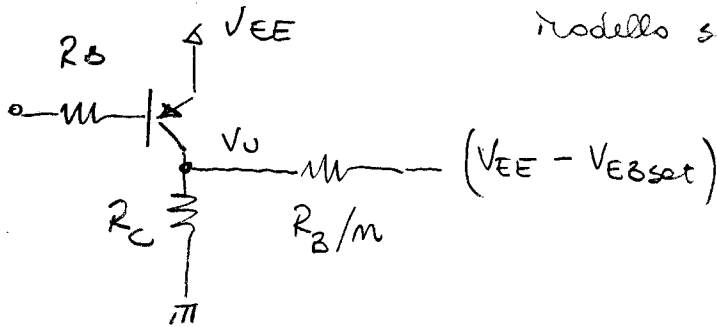
Progettare una rete di Moore con 2 ingressi e una uscita in grado di riconoscere l'uguaglianza di terne di valori consecutivi ai 2 ingressi. La rete riconosce l'uguaglianza (anche per terne interallacciate) ponendo l'uscita a 1 per un ciclo di clock.

## **ESERCIZIO N°5**

8/5 punti

Realizzare un programma per un microcontrollore a scelta della famiglia AVR in grado di emulare uno shift register di 36 bit. L'ingresso CK (clock) corrisponde al pin PB0, l'ingresso SI (serial in) corrisponde a PB1 e l'uscita SO (serial out) corrisponde al pin PB2. Nel caso si adotti un microcontrollore XMEGA, configurare la periferica opportuna in modo che i pin indicati appartengano alla porta virtuale 0.

1



Sul livello alto si ha  $NMH = 0,6$ , indipendente da  $n$  in quanto i bipolari a valle sono OFF.

Sul livello basso la  $V_O$  deve restare sotto a

$$V_{IL} - NML = V_{EE} - V_{EBset} - R_B \cdot \frac{V_{EE} - V_{EBset} - NMH}{\beta_{FE} R_C}$$

$$V_O < 7 - 0,8 - \frac{15}{240} 6,9 - 0,6 = 5,169 \text{ V}$$

Quindi

$$\frac{R_C}{R_C + R_B/m} (V_{EE} - V_{EBset}) \leq V_{OJMAX} \quad \text{da cui}$$

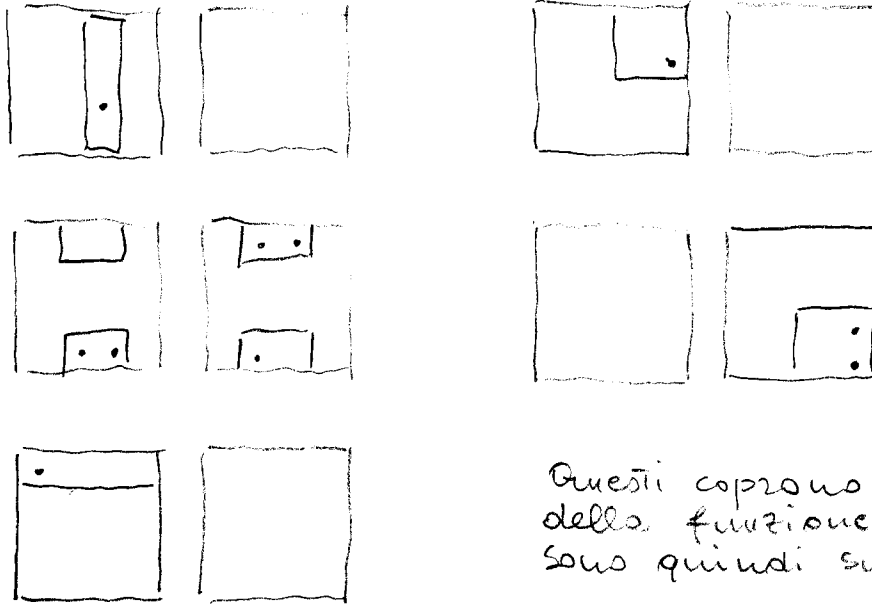
$$m \leq 62$$

2

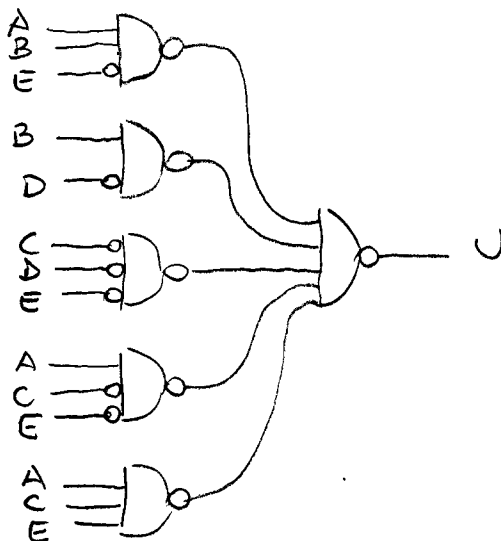
Ricavare la mappa della funzione

|    |    | E=0 |    |    |    | E=1 |    |    |    |    |    |
|----|----|-----|----|----|----|-----|----|----|----|----|----|
|    |    | AB  |    |    |    | AB  |    |    |    |    |    |
| CD | AB | 00  | 01 | 11 | 10 | CD  | AB | 00 | 01 | 11 | 10 |
|    | 00 | 1   | 1  | 1  | 1  | 0   | 0  | 1  | 1  | 0  | 0  |
| 01 | 0  | 0   | 1  | 1  | 0  | 0   | 0  | 0  | 0  | 0  | 0  |
| 11 | 0  | 0   | 1  | 0  | 0  | 0   | 1  | 1  | 0  | 0  | 1  |
| 10 | 0  | 1   | 1  | 0  | 0  | 0   | 1  | 1  | 1  | 0  | 1  |

La sintesi ottima NAND-NAND si ricava dalla SP  
Trovo gli impicanti principali essenziali



Questi coprono tutti gli 1 della funzione. Sono quindi sufficienti.



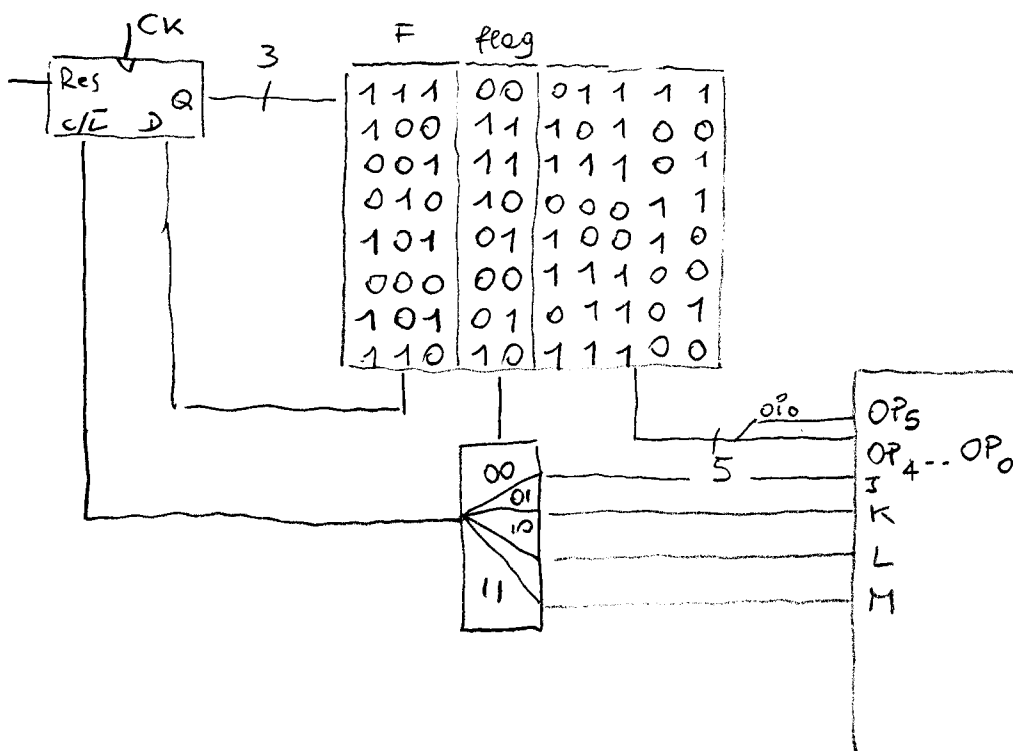
③ Riordino gli stati in modo da avere una sequenza ciclica completa (o vero)

| (cod) | stato | if | then | op     |
|-------|-------|----|------|--------|
| 000   | A     | J  | F    | 101111 |
| 001   | D     | M  | H    | 010100 |
| 010   | E     | M  | D    | 111101 |
| 011   | B     | L  | E    | 100011 |
| 100   | H     | K  | G    | 010010 |
| 101   | G     | J  | A    | 011100 |
| 110   | C     | K  | G    | 101101 |
| 111   | F     | L  | C    | 011100 |

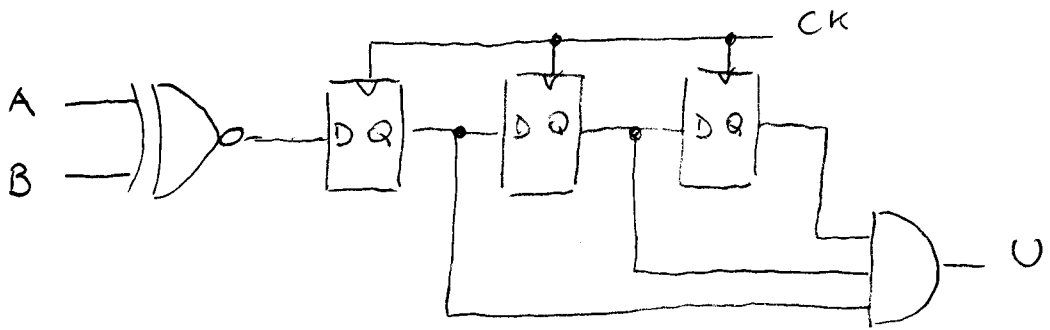
osservo che  
 $OP_0 = OP_5$

|   |    |
|---|----|
| J | 00 |
| K | 01 |
| L | 10 |
| M | 11 |

Schemo-



④



# 5

```
LDI R16,0x01
STS GPIO+VPCTRLA,R16 //definisce VPORT0 come PORTB
SBI VPORT0+DIR,2 //mette PB2 (S0) in uscita
loop:
SBIS VPORT0+IN,0 //aspetta PB0 (CK) up
RJMP loop
CLC //clear Carry
SBIC VPORT0+IN,1 //guarda PB1 (SI)
SEC //set Carry
ROR R0
ROR R1
ROR R2
ROR R3
ROR R4 //esegue lo shift
SBRC R4,3 //guarda il bit da far uscire
RJMP out1
out0:
CBI VPORT0+OUT,2
RJMP loop1
out1:
SBI VPORT0+OUT,2
loop1:
SBIC VPORT0+IN,0 //aspetta PB0 (CK) down
RJMP loop1
RJMP loop //loop perpetuo
```