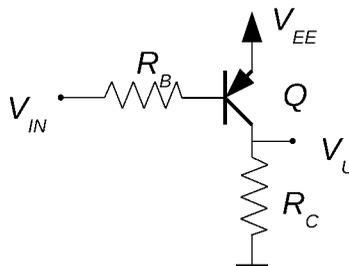


ESERCIZIO N°1

7/4 punti

Determinare la caratteristica di trasferimento a vuoto della seguente porta RTL con transistor *pnp* e individuare V_{IL} , V_{IH} , V_{OL} , V_{OH} . Individuare infine il valore massimo della corrente di uscita che può essere erogato dalla porta senza che la tensione di uscita si riduca apprezzabilmente.

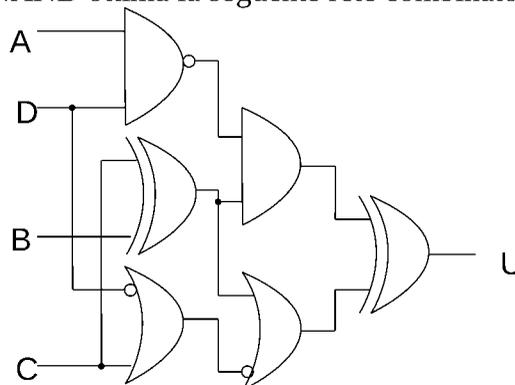
$R_B = 5 \text{ k}\Omega$; $R_C = 500 \text{ }\Omega$; $V_{EE} = 5 \text{ V}$; $h_{FE} = 100$.



ESERCIZIO N°2

6/4 punti

Realizzare in forma NAND-NAND ottima la seguente rete combinatoria a 4 ingressi e una uscita.



ESERCIZIO N°3

6/4 punti

Spiegare la differenza tra le istruzioni MUL e FMUL del linguaggio assembly della famiglia AVR ed esprimere in binario il valore del risultato R1:R0 delle due diverse istruzioni e il valore dei flag quando gli argomenti sono due registri il cui contenuto è rispettivamente 0xEC e 153.

ESERCIZIO N°4

6/4 punti

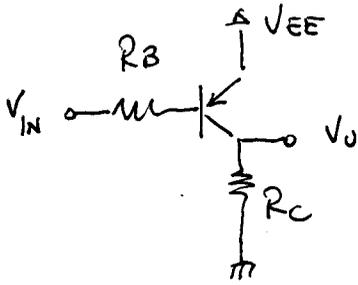
Progettare una rete di Moore a un ingresso (A) e 3 uscite (Q_2 , Q_1 e Q_0) in grado di generare una sequenza in cui il valore successivo è pari al valore precedente più 3, se l'ingresso è 0, e invece è pari al valore precedente meno 1 se l'ingresso è 1. Le somme e le differenze si intendono modulo 8.

ESERCIZIO N°5

8/5 punti

Nella memoria dati di un microcontrollore AVR XMEGA256A3BU, a partire dall'indirizzo 0x3000 sono contenuti i 4 valori interi con segno v_1 , v_2 , v_3 e v_4 , da 8 byte ciascuno, memorizzati a partire dal byte meno significativo. Scrivere un sottoprogramma che fa scorrere tali numeri, replicando l'ultimo. Dopo una chiamata al sottoprogramma richiesto quindi a partire da 0x3000 ci saranno i valori v_2 , v_3 , v_4 e ancora v_4 .

1



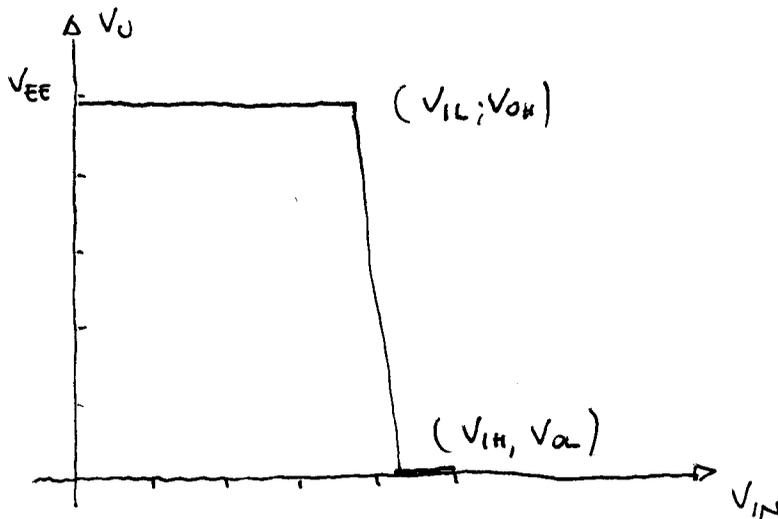
$$V_{OL} = \phi \quad (\text{pmp interdetto})$$

$$V_{IL} = V_{EE} - V_{EBon} = 4,3 \text{ V}$$

$$V_{OH} = V_{EE} - V_{ECsat} = 4,9 \text{ V}$$

$$V_{IH} = V_{EE} - V_{EBsat} - R_B \frac{V_{OH}}{h_{FE} R_C}$$

$$= 3,71 \text{ V}$$



Si ha una variazione significativa dell'uscita sul livello alto quando il BJT esce dalla saturazione. Quindi, con $V_{IN} = 0$,

$$I_U + \frac{V_{OH}}{R_C} = h_{FE} \frac{V_{EE} - V_{EBsat}}{R_B} \quad \text{da cui}$$

$$I_U = 74,2 \text{ mA}$$

② La funzione \bar{U} , con semplici manipolazioni booleane

$$U = [(B \oplus C) \cdot (\bar{A} + \bar{D})] \oplus [(B \oplus C) + \bar{C}D]$$

Ricavo la mappa a partire dalle mappe dei due termini delle XOR finali

		AD			
Bc		00	01	11	10
0	00	0	0	0	0
1	01	1	1	0	1
0	11	0	0	0	0
1	10	1	1	0	1

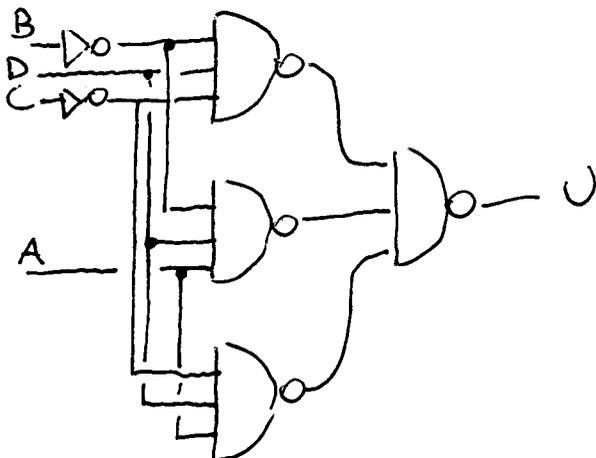
(B ⊕ C)

		AD			
Bc		00	01	11	10
00		0	1	1	0
01		1	1	1	1
11		0	0	0	0
10		1	1	1	1

		AD			
Bc		00	01	11	10
00		0	1	1	0
01		0	0	1	0
11		0	0	0	0
10		0	0	1	0

Sintesi ottimale SP (da cui NAND-NAND)

$$\bar{B}\bar{C}D + AD\bar{B} + AD\bar{C} = U$$



③

MUL opera con interi assoluti.
pone il risultato in R1:R0, senza segno su 16b

FMUL opera su frazionali 1.7 senza segno.
il risultato, senza segno, è posto in R1:R0
nel formato 1.15

Per ottenerlo occorre scorrere di 1 bit a sinistra
il risultato della MUL. (che sarebbe 2.14)

$$R_d = 0 \times EC \quad (236)$$

$$R_2 = 153$$

MUL R_d, R_2 in R1:R0 $\leftarrow 36108$

1000110100001100

$$C = 1 \quad (R_{15})$$

$$Z = 0$$

FMUL R_d, R_2

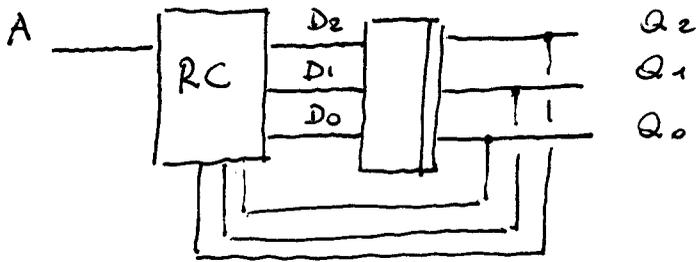
0001101000011000

$$C = 1 \quad (R_{16})$$

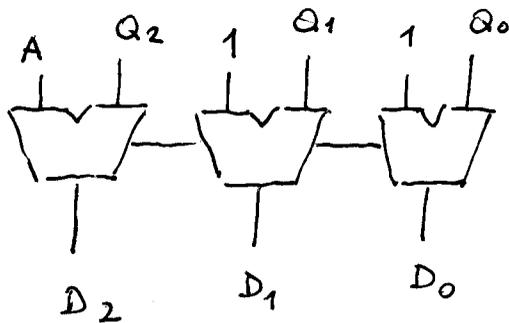
$$Z = 0$$

④

La rete ha la seguente architettura (facendo coincidere stato presente e uscite)



La rete RC può essere realizzata con un sommatore che somma 3 o 7 (sottrae 1 modulo 8), su controllo di A



5

```
rpn_stack_down:
push R16
push R17
push XL
push XH
push YL
push YH
ldi XL,low(v1)
ldi XH,high(v1)
ldi YL,low(v2)
ldi YH,high(v2)
ldi R16,8*3 //valori da copiare
loop:
    ld R17,Y+
    st X+,R17
    dec R16
    brne loop
pop YH
pop YL
pop XH
pop XL
pop R17
pop R16
ret
```

```
/* Questa parte permette di usare nomi simbolici al posto
dei valori costanti */
.dseg //direttiva che indica la memoria dati
.org 0x3000 //direttiva che specifica l'indirizzo
v1:
.byte 8 //direttiva che alloca 8 byte per una variabile
v2:
.byte 8
v3:
.byte 8
v4:
.byte 8
```