

Cognome

Nome

ESERCIZIO N°1

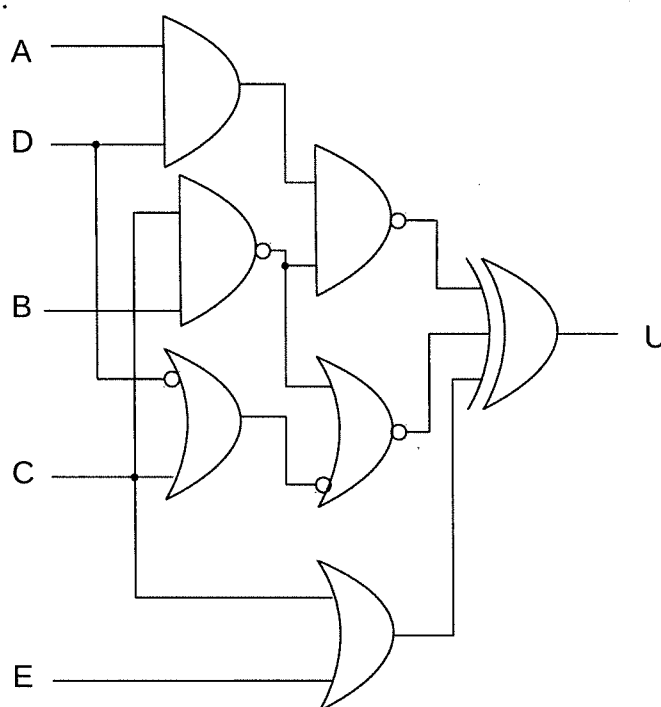
8 punti

Realizzare una subroutine per il microcontrollore AVR XMEGA256A3BU, che lascia in R16 la parte intera della radice quadrata della singola cella puntata da X (intesa come numero senza segno). Determinare il tempo massimo di esecuzione della subroutine proposta.

ESERCIZIO N°2

5 punti

Esprimere in forma SP a minimo numero di letterali la funzione combinatoria U corrispondente al seguente schema logico.



ESERCIZIO N°3

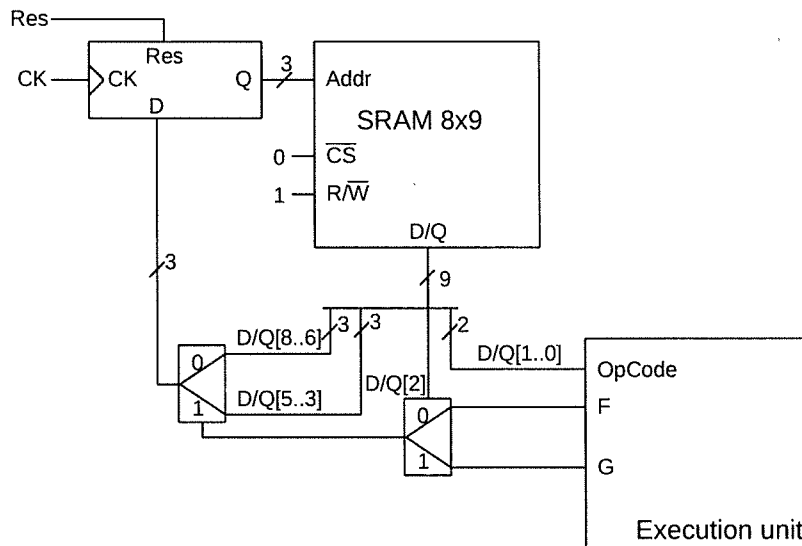
4 punti

Disegnare lo schema logico di un contatore up modulo 10 con abilitazione, usando T-FF e porte logiche.

ESERCIZIO N°4

5 punti

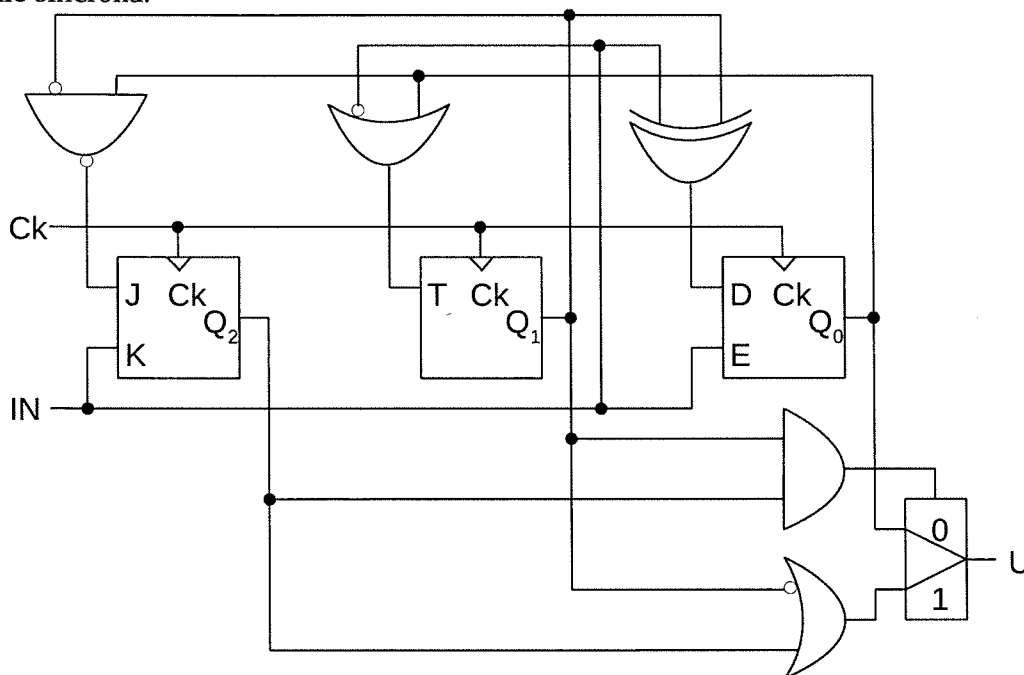
Determinare il diagramma di flusso, attribuendo agli stati un nome a scelta, del seguente sequenziatore. Le 8 righe della SRAM contengono i seguenti 8 valori: 71, 461, 150, 432, 484, 37, 155, 234. Sarebbe stato possibile realizzare il sequenziatore con un contatore a caricamento parallelo, risparmiando sulla dimensione della SRAM (spiegare)?



ESERCIZIO N°5

5 punti

Individuare la tipologia architetturale e disegnare il grafo delle transizioni della seguente macchina sequenziale sincrona.



ESERCIZIO N°6

6 punti

Siano dati i 3 valori positivi $1,28 \cdot 10^{-4}$; $1,57 \cdot 10^{-4}$; $4,51 \cdot 10^3$

- Determinare la rappresentazione in virgola mobile IEEE754-2008 (binary32) dei 3 numeri (con arrotondamento al numero di macchina più vicino).
- Valutare la rappresentazione della somma tra i 3 numeri rappresentati, eseguita in tutti i modi possibili, usando 2 volte un unico sommatore ideale a 2 ingressi, con rappresentazione dell'uscita in formato binary32 (usare l'arrotondamento classico se necessario).

1

Realizzare una subroutine per il microcontrollore AVR XMEGA256A3BU, che lascia in R16 la parte intera della radice quadrata della singola cella puntata da X (intesa come numero senza segno). Determinare il tempo massimo di esecuzione della subroutine proposta.

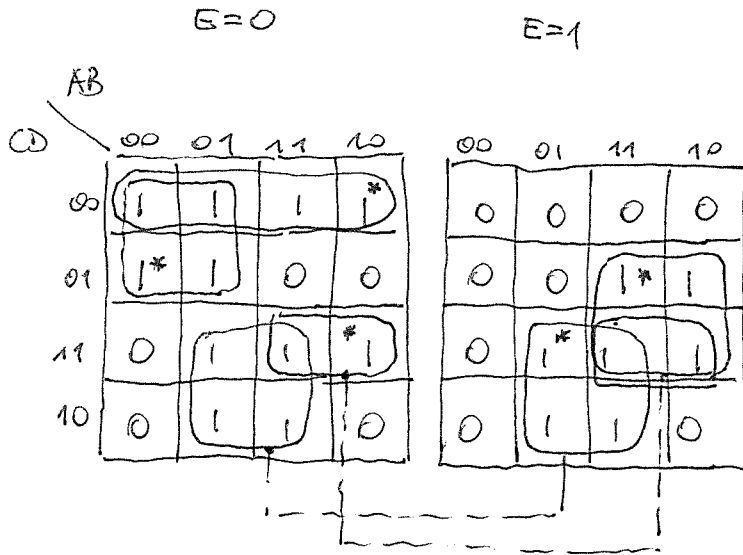
```
root8:
    push R17
    push R1
    push R0
    ld R17,X //carica il dato da valutare
    ldi R16,16 //algoritmo per tentativi
loop:
    dec R16 //parte da 15
    mul R16,R16
    cp R17,R0
    brcs loop //quando esce ha trovato la radice
pop R0
pop R1
pop R17
ret //115 cicli con il dato nullo (caso peggiore) compresa rcall e ret
```

② Ricavo l'espressione di U

$$U = (E+C) \oplus (\overline{AD} \overline{CB}) \oplus [BC \cdot (C+\overline{D})] \quad \text{con Shannon su C}$$

$$\begin{aligned} U &= C \cdot (A\overline{D}\overline{B} \oplus B) + \overline{C} (E \oplus \overline{AD}) = \\ &= CB + C\overline{B}AD + \overline{C}E \cdot AD + \overline{C}E\overline{A} + \overline{C}E\overline{D} \end{aligned}$$

Mappe



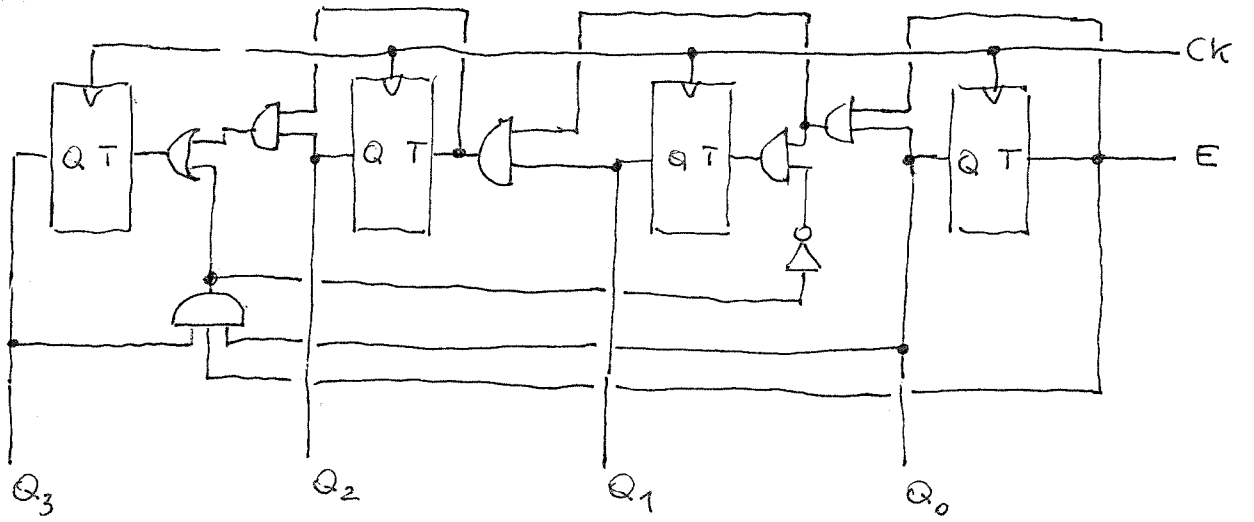
Con * si evidenziano i mintermini che rendono ESSENZIALE il relativo implicante di appartenenza

ESPRESSIONE SP e minimo numero di letterali

$$U = \overline{C}\overline{D}\overline{E} + \overline{A}\overline{C}\overline{E} + ACD + BC + ADE \quad (14 \text{ letterali})$$

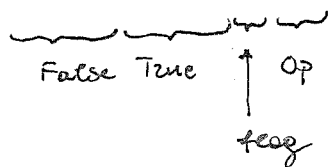
3

...
1 0 0 1
- 1 0 1 0 -
- 0 0 0 0 -
...
F - B -



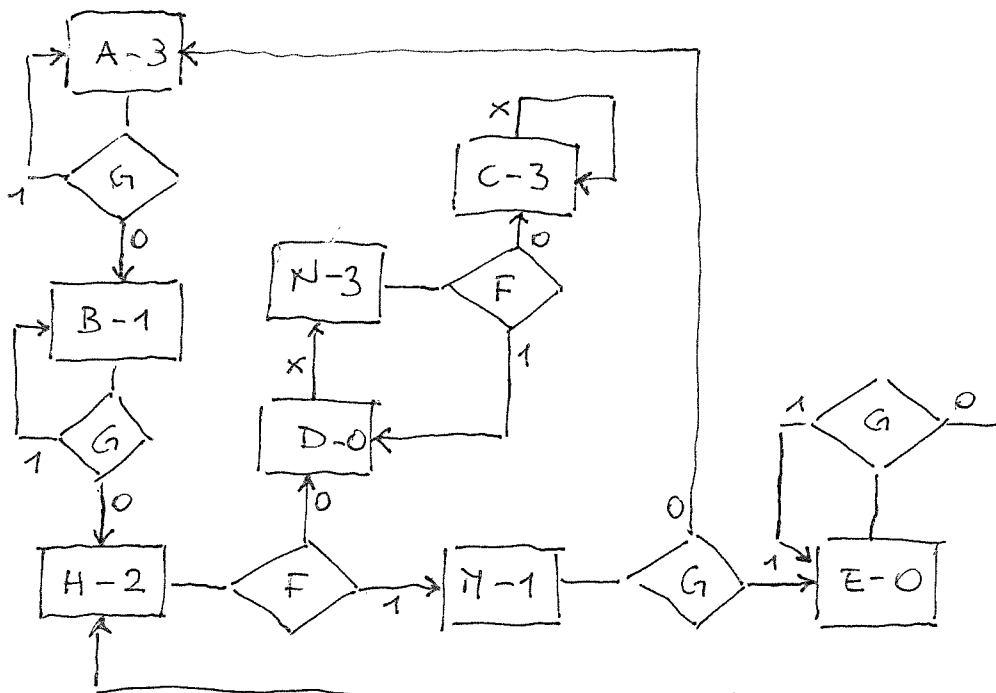
④ Della SRAM (contenuto convertito in binario) ricavo il codice e quindi il diagramma

Addr	Cont.	Codifica	Flag
000:	001000111	000 A	
001:	111001101	001 B	0 F
010:	010010110	010 C	1 G
011:	110110000	011 D	
100:	111100100	100 E	
101:	000100101	101 M	
110:	010011011	110 N	
111:	011101010	111 H	



A : if G then A else B; op 3
 B : if G then B else H; op 1
 C : if G then C else C; op 2
 D : if F then N else N; op 0
 E : if G then E else H; op 0
 M : if G then E else A; op 1
 N : if F then D else C; op 3
 H : if F then M else D; op 2

il sequenziatore non può essere realizzato con contatore in quanto non esiste SEQUENZA CICLICA COMPLETA (né con flag vero né falso)



5

L'uscita dipende SOLO dallo stato.
 la macchina è di Moore

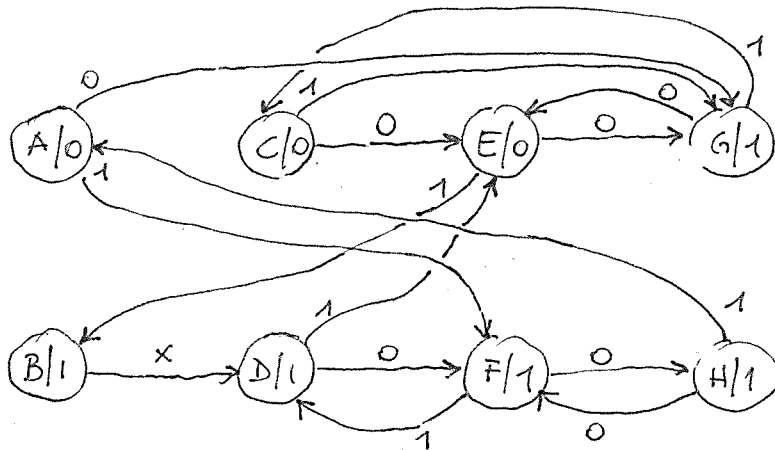
Rete per l'uscita $U = \overline{Q_1} \cdot \overline{Q_2} \cdot Q_0 + Q_1 Q_2 (\overline{Q_1} + \overline{Q_2}) = \overline{Q_1} Q_0 + \overline{Q_2} Q_0 + Q_1 Q_2$

Reti per i FF: $J = Q_1 + \overline{Q_0}$ $K = \overline{Q_1}$; $T = \overline{Q_1} + Q_0$;

$D = \overline{Q_1} \oplus Q_1$ $E = \overline{Q_1}$

Tabella di eccitazione (e transizione)

Code	Q_2	Q_1	Q_0	$\overline{Q_0}$	J	K	Q_2^+	T	Q_1^+	D	E	Q_0^+	SF	U
A	0	0	0	0	1	0	1	1	1	0	0	0	S	0
				1	1	1	0	0	1	1	1	F		
B	0	0	1	0	0	0	0	1	1	0	0	1	D	1
				1	0	1	0	1	1	1	1	D		
C	0	1	0	0	1	0	1	1	0	1	0	0	E	0
				1	1	1	0	1	0	1	0	G		
D	0	1	1	0	1	0	1	1	0	0	1	1	F	1
				1	1	1	1	0	0	1	0	E		
E	1	0	0	0	1	0	1	1	1	0	0	0	G	0
				1	1	1	0	0	1	1	1	B		
F	1	0	1	0	0	0	1	1	1	0	0	1	H	1
				1	0	1	0	1	1	1	1	D		
G	1	1	0	0	1	0	1	1	0	1	0	0	E	1
				1	1	1	0	0	1	0	1	C		
H	1	1	1	0	1	0	1	1	0	1	0	1	F	1
				1	1	1	0	1	0	0	1	A		



$$\textcircled{b} \quad x = (-1)^5 \cdot 2^{(E-127)} \cdot (1 + T \cdot 2^{-23})$$

$$a \quad 1,28 \cdot 10^{-4} = (-1)^0 \cdot 2^{-13} \cdot (1 + 407485,02... \cdot 2^{-23})$$

$$0 \mid 01110010 \mid 0000110001101111011110111101 \mid$$

$$b \quad 1,57 \cdot 10^{-4} = (-1)^0 \cdot 2^{-13} (1 + 2400349,84... \cdot 2^{-23})$$

$$0 \mid 01110010 \mid 01001001010000001011110 \mid$$

$$c \quad 4,51 \cdot 10^3 = (-1)^0 \cdot 2^{12} (1 + 847872 \cdot 2^{-23})$$

$$0 \mid 10001011 \mid 000110011110000000000000 \mid$$

Se si somma \hat{a} partire da \hat{c} , il risultato (riportando il parziale in notazione IEEE 754) sarà ancora \hat{c} poiché

$$\begin{aligned} \hat{c} + \hat{a} &= \hat{c} & (\text{ci sono 25 olog binari}) \\ \hat{c} + \hat{b} &= \hat{c} \end{aligned}$$

Se si esegue prima $\hat{a} + \hat{b}$ avremo un aumento di olog binario

$$0 \mid 01110011 \mid 0010101011011000001110 \mid$$

questo modificherebbe, a causa dell'arrotondamento, la somma

$$\hat{s} = (\hat{a} + \hat{b}) + \hat{c}$$

$$0 \mid 10001011 \mid 000110011110000000000001 \mid$$