

ESERCIZIO N°1

8 punti

Realizzare una subroutine per il microcontrollore AVR XMEGA256A3BU, che lascia in Z un numero che esprime quante volte il codice 0xAA è contenuto complessivamente in memoria (cioè tra gli indirizzi 0x2000 e 0x5FFF, compresi gli estremi).

ESERCIZIO N°2

5 punti

Esprimere in forma NAND-NAND a minimo numero di letterali la funzione combinatoria U corrispondente alla seguente espressione:

$$[(A+\bar{B}+C)(\bar{D}+E)(\bar{A}+\bar{E}+B)(A+B+\bar{C}+\bar{D}+E)]\oplus[(E+\bar{B}+D+\bar{A})(\bar{D}+C+B)(\bar{E}+\bar{C}+\bar{D})]$$

ESERCIZIO N°3

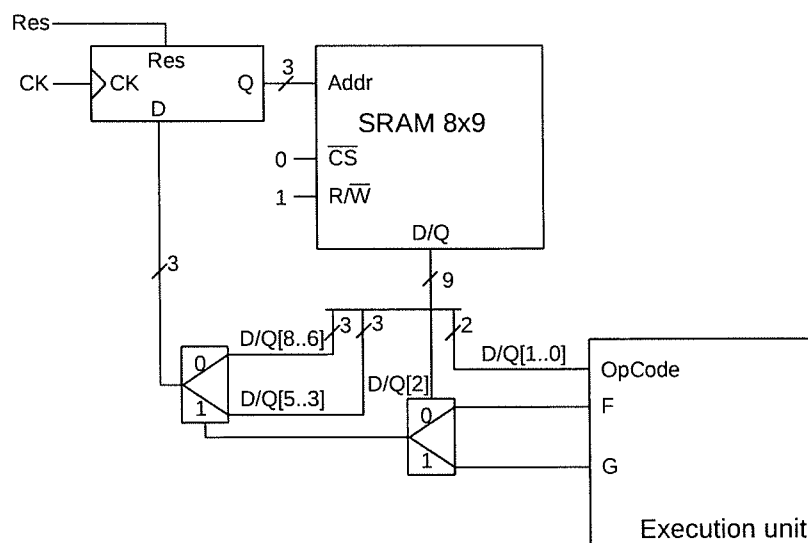
5 punti

Disegnare lo schema logico di un contatore Johnson modulo 9 con abilitazione, usando DE-FF e porte logiche. Progettare una rete combinatoria che usa le 5 uscite dei DE-FF per produrre la sequenza ciclica 010111010.

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: 428, 372, 53, 511, 181, 237, 122, 331. Sarebbe stato possibile realizzare il sequenziatore con un contatore a caricamento parallelo, risparmiando sulla dimensione della SRAM (spiegare)?



ESERCIZIO N°5

5 punti

Il registro R16 di un microcontrollore AVR XMEGA256A3BU contiene il valore 0xBA, mentre il puntatore Z vale 0x23FF e a questo indirizzo è stato inserito il valore 87. Determinare il contenuto (in esadecimale) di R16 e della cella all'indirizzo 0x23FF dopo che è stata eseguita ciascuna delle seguenti istruzioni (a partire dalle condizioni assegnate):

XCH Z, R16; LAC Z, R16; LAT Z, R16; LAS Z, R16.

ESERCIZIO N°6

5 punti

Siano dati i 4 numeri (A, B, C, D) la cui rappresentazione IEEE754-2008 (binary32) è costituita dai seguenti valori espressi in esadecimale 0x739AD451, 0x73BB593A, 0x73D1075F, 0x738D38C2.

- a) Determinare i 4 numeri in notazione scientifica con 7 cifre significative.
- b) Valutare (commentando) la rappresentazione della somma tra i 4 numeri rappresentati, eseguendo in un caso $(A + B) + (C + D)$ e nell'altro $(A + C) + (B + D)$, usando 3 volte un unico sommatore ideale a 2 ingressi, con rappresentazione dell'uscita in formato binary32 (che usa l'arrotondamento classico).

1

Realizzare una subroutine per il microcontrollore AVR XMEGA256A3BU, che lascia in Z un numero che esprime quante volte il codice 0xAA è contenuto complessivamente in memoria (cioè tra gli indirizzi 0x2000 e 0x5FFF compresi gli estremi).

```
findAA:
    push XL
    push XH
    push R16
    push R17
    ldi R17,0xAA
    ldi XL,low(0x2000)
    ldi XH,high(0x2000)
    ldi ZL,low(0x4000) //massimo valore possibile
    ldi ZH,high(0x4000)
loop:
    ld R16,X+
    cpse R16,R17
    sbiw ZH:ZL,1 //decrementa se diversi
    cpi XL,low(0x6000)
    brne loop
    cpi XH,high(0x6000)
    brne loop
pop R17
pop R16
pop XH
pop XL
ret
```

2

Ricavo le mappe delle funzioni U , considerando i due membri della XOR

$E=0$

		AB			
		00	01	11	10
CD	00	1	0	1	1
	01	0	0	0	0
	11	0	0	0	0
	10	1	1	1	1

		AB			
		00	01	11	10
CD	00	1	0	1	0
	01	1	0	1	0
	11	1	1	1	0
	10	1	1	1	0

1° membro

		AB			
		00	01	11	10
CD	00	1	1	0	1
	01	0	1	1	0
	11	1	1	1	1
	10	1	1	0	1

		AB			
		00	01	11	10
CD	00	1	1	1	1
	01	0	1	1	0
	11	0	0	0	0
	10	1	1	1	1

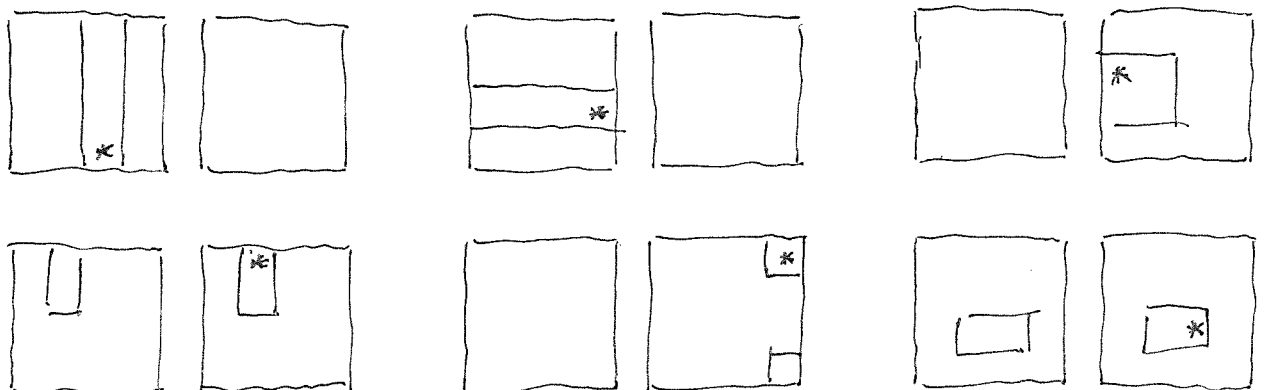
2° membro

		AB			
		00	01	11	10
CD	00	0	1	1	0
	01	0	1	1	0
	11	1	1	1	1
	10	0	0	1	0

		AB			
		00	01	11	10
CD	00	0	1	0	1
	01	1	1	0	0
	11	1	1	1	0
	10	0	0	0	1

U

Per realizzare la forma NAND-NAND si parte da una forma normale SP. implicanti essenziali:

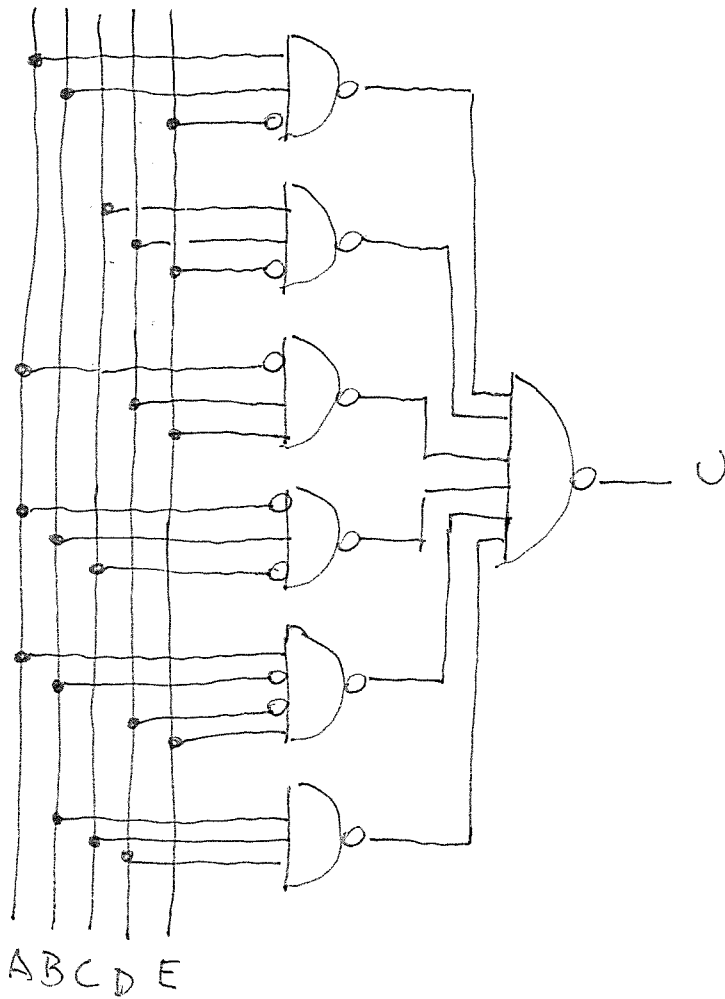


Coprons TUTTI i mintermini. Quindi:

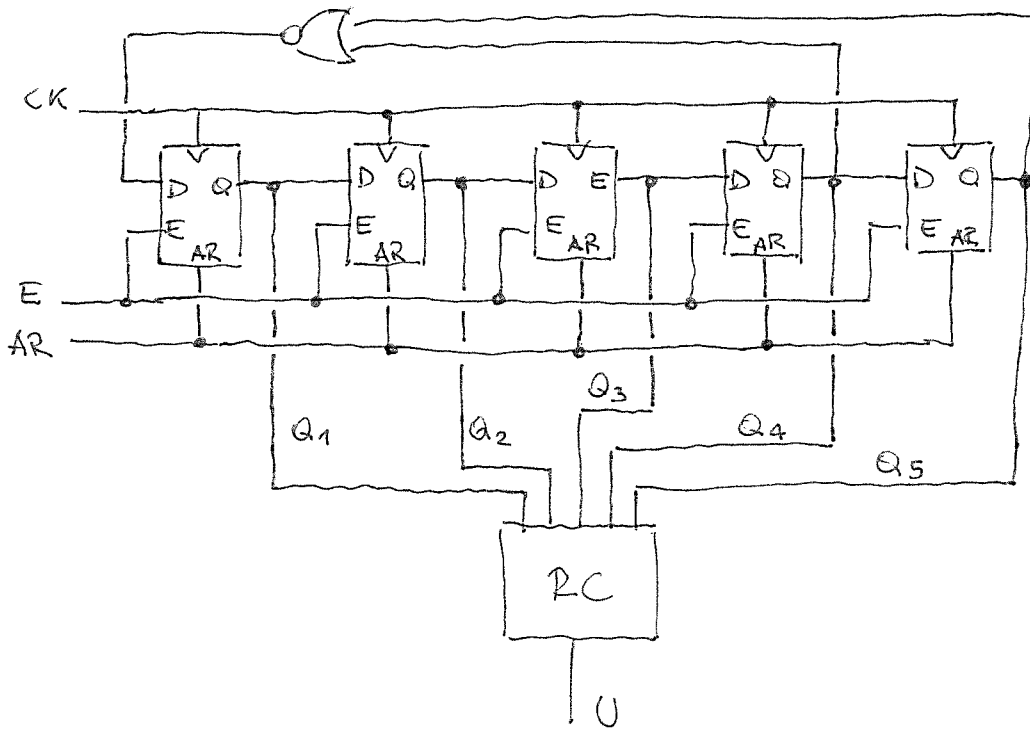
Forme NAND-NAND

$$U = \overline{AB\bar{E}} \cdot \overline{CDE} \cdot \overline{ADE} \cdot \overline{ABC} \cdot \overline{ABDE} \cdot \overline{BCD}$$

Scheme (non richiesto)



③ Contatore Johnson mod 9



Successione dei valori

	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	U
g	0	0	0	0	0	0
	1	0	0	0	0	1
	1	1	0	0	0	0
	1	1	1	0	0	1
	1	1	1	1	0	1
	0	1	1	1	1	1
	0	0	1	1	1	0
	0	0	0	1	1	1
	0	0	0	0	1	0
	0	0	0	0	0	0

		Q ₁ Q ₂							
Q ₃ Q ₄		00	01	11	10	00	01	11	10
	00		0	-	0	1	0	-	-
01		1	-	-	1	1	-	-	1
11		-	1	1	-	0	1	-	-
10		-	-	1	1	-	-	-	1

$Q_5 = \phi$

$Q_5 = 1$

$U = Q_1 \bar{Q}_2 + \bar{Q}_3 Q_4 + Q_2 Q_3$

④ Troviamo il MICROCODICE

S ₀ :	IF	G	THEN	S ₅	ELSE	S ₆
S ₁ :		G		S ₆		S ₅
S ₂ :		G		S ₆		S ₀
S ₃ :		G		S ₇		S ₇
S ₄ :		G		S ₆		S ₂
S ₅ :		G		S ₅		S ₃
S ₆ :		F		S ₇		S ₁
S ₇ :		F		S ₁		S ₅

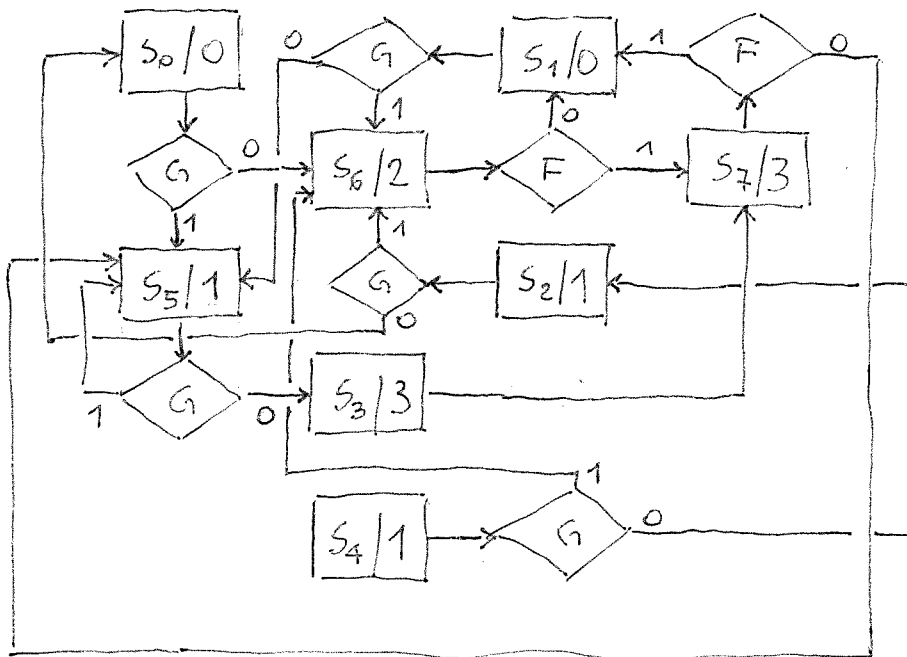
SRAM

1	1	0	1	0	1	1	0	0
1	0	1	1	1	0	1	0	0
0	0	0	1	1	0	1	0	1
1	1	1	1	1	1	1	1	1
0	1	0	1	1	0	1	0	1
0	1	1	1	0	1	1	0	1
0	0	1	1	1	1	0	1	0
1	0	1	0	0	1	0	1	1

F | V | | Op

Non si può fare con il contatore perché non esiste sequenza COMPLETA (ogni colonna presenta ripetizioni)

0: flag F
1: G



Note: S₄ non viene mai raggiunto

5

situazione. Il valore di Z 0x23FF appartiene alla SRAM quindi le operazioni seguenti

R16	hex	(Z)	hex	
10111010	BA	01010111	57	Valore INIZIALE
01010111	57	10111010	BA	Dopo XCH
"	57	01000101	45	Dopo LAC
"	57	11001101	ED	Dopo LAT
"	57	11111111	FF	Dopo LAS

⑥ legge di rappresentazione dei numeri NORMALIZZATI

$$x = (-1)^S \cdot 2^{E-127} (1 + T \cdot 2^{-23})$$

Eseminiamo i numeri

A: 0|11100111|0011010110101000101010001

S=0 E=231 T=1758289

A = + 2,453369 · 10³¹

B: 0|11100111|01110110101100100100111010

S=0 E=231 T=3889466

B = + 2,968556 · 10³¹

C: 0|11100111|1010001000001110101111

S=0 E=231 T=5310303

C = + 3,312193 · 10³¹

D: 0|11100111|00011010011100011000010

S=0 E=231 T=866498

D = + 2,237747 · 10³¹

La somma (A+C)+(B+D) verrà eseguita esattamente, in quanto l'eliminazione del bit meno significativo conseguente all'aumento dell'esponente di 2 non porta ERRORE DI ARROTONDAMENTO.

S=0 E=233 T=2956139

Nel caso invece di (A+B)+(C+D) le prime due somme introducono ciascuna arrotondamento e anche l'ultima somma richiede arrotondamento. Si ottiene

S=0 E=233 T=2956140

con 1 LSB in più, la somma si conferma NON ASSOCIATIVA