

ESERCIZIO N°1

5 punti

Determinare i bit della rappresentazione binary 32 IEEE 754-2008 con la migliore approssimazione possibile dei seguenti numeri, ove M è la matricola dello studente:

 $M, (3M)^6, -(11M)^{-6}$

e valutare per ciascuno l'errore relativo $\frac{\hat{x} - x}{x}$

ESERCIZIO N°2

8 punti

Realizzare una subroutine ritardo per il microcontrollore AVR XMEGA256A3BU che non altera alcun registro (compreso CPU_SREG) e viene eseguita in M cicli di clock, da RCALL a RET compresi.

ESERCIZIO N°3

5 punti

Progettare e disegnare lo schema logico di una rete sequenziale sincrona con un ingresso di abilitazione E (se $E = 0$ lo stato della macchina non cambia) in grado di generare una sequenza periodica il cui periodo vale $\{b_{11}, \bar{b}_5, b_9, b_7, \bar{b}_7, b_5, b_0\}$. I valori b_0, b_1, \dots sono le cifre binarie di M ; b_0 è il bit meno significativo.

ESERCIZIO N°4

5 punti

Lo studente proponga una funzione combinatoria a scelta con 5 variabili di ingresso X_4, X_3, X_2, X_1, X_0 , nella cui tabella di verità siano presenti 13 "1", 11 "0" e 8 "-". La funzione deve avere 2 e 2 soli implicanti principali di ordine 3 (compresi i don't care considerati come "1"), oltre ad altri implicanti di ordine minore e ovviamente nessuno di ordine maggiore. Sintetizzare la funzione in forma PS ottima (minimo numero di letterali), indicando esplicitamente quali sono gli implicati essenziali (non è richiesto il disegno dello schema logico).

ESERCIZIO N°5

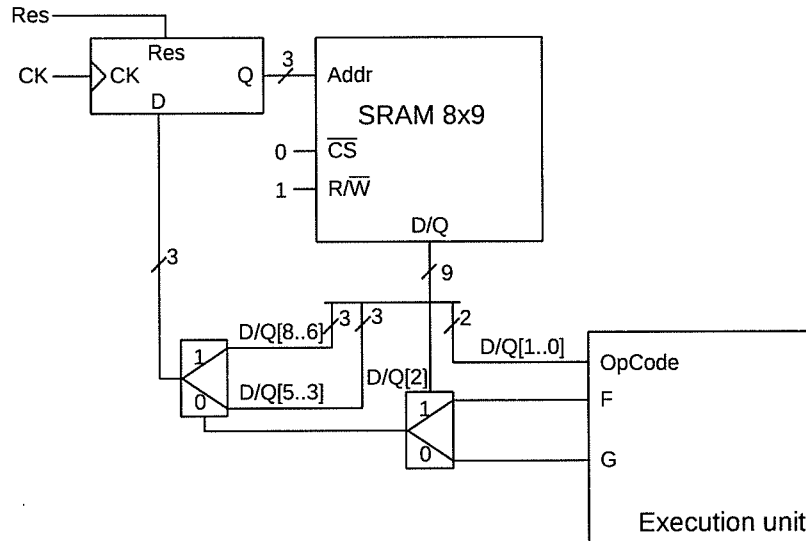
5 punti

Lo studente realizzi la funzione dell'esercizio precedente avendo a disposizione invertitori e 12 multiplexer 2:1.

ESERCIZIO N°6

5 punti

Determinare il diagramma di flusso del seguente sequenziatore. Agli stati deve essere attribuito il nome S_i , con i corrispondente al codice dello stato. Le 8 righe della SRAM contengono i seguenti 8 valori esadecimali: 1AC, 15A, 1E8, 14E, 10D, 1ED, 129, 05A. Sarebbe stato possibile realizzare il sequenziatore con un contatore a caricamento parallelo, risparmiando sulla dimensione della SRAM (spiegare)?



①

Determinare i bit delle rappresentazioni binary 32 di
(a titolo di esempio, $M = 500000$)

M essendo minore di 2^{24} , avrà una rappresentazione
ESATTA

$$S = \emptyset \quad E = 145 \quad (0x91) \quad T = 7611392 \quad (0x742400)$$

$$0 | 10010001 | 111010000100100000000000 |$$

$$E_2 = \emptyset$$

$(3M)^6$ è vicino al limite superiore di rappresentabilità, ma
è ancora rappresentabile senza problemi

$$S = \emptyset \quad E = 250 \quad (0xFA) \quad T = 597011 \quad (0x091C13)$$

$$0 | 11111010 | 00010010001110000010011 |$$

$$E_2 = 2,7985 \cdot 10^{-8}$$

$-(11M)^{-6}$ è invece minore del più piccolo dei normalizzabili
(in modulo), quindi va rappresentato con la legge
dei non norm.

$$\begin{cases} x_{norm} = (-1)^S \cdot 2^{E-127} \cdot (1 + T \cdot 2^{-23}) \\ x_{nonorm} = (-1)^S \cdot 2^{-149} \cdot T \quad (\text{con } E = \emptyset) \end{cases}$$

$$S = 1 \quad E = 0 \quad T = 25781 \quad (0x0064B5)$$

$$0 | 00000000 | 000000001110010010110101 |$$

$$E_2 = 1,5063 \cdot 10^{-5}$$

2

Realizzare una subroutine ritardo per il microcontrollore AVR XMEGA256A3BU che non altera alcun registro (compreso CPU_SREG) e viene eseguita in M cicli di clock, da CALL (4 cicli) a RET (5 cicli) compresi.

```
ritardo:
.equ k1=0x10 //valori che garantiscono il ritardo richiesto
.equ k2=0x00
.equ k3=0xDD
    push R16 //registro di appoggio per il salvataggio di CPU_SREG
    lds R16,CPU_SREG
    push R16 //registri di appoggio per il ritardo
    push R17
    push R18
    ldi R16,k1 //primo byte
    ldi R17,k2
    ldi R18,k3
loop:
    dec R16 //valore iniziale k1 poi 0 alle successive iterazioni
    brne loop
    dec R17 //valore iniziale k2 poi 0 alle successive iterazioni
    brne loop
    dec R18
    brne loop
    nop // x nop per tuning fine (0<=x<=2)
    nop
    pop R18
    pop R17
    pop R16
    sts CPU_SREG,R16 //ripristina flag
    pop R16
    ret
```

```
/*
calcolo dei coefficienti k1, k2 e k3
ritardo complessivo in cicli di clock
```

```
9 rcall e ret
12 push e pop
4 lds e sts
3 ldi per inizializzazione
x nop eventuali
```

```
ritardo del loop composito (hp: k1, k2, k3 diversi da 0)
 $3*k1-1+3+(3*256-1+3)*(k2-1)-1+3+((3*256-1+3)*256-1+3)*(k3-1)=$ 
 $=3*(k1+256*k2+256^2*k3)-197888$ 
```

```
sommando tutto
 $R=3*(k1+256*k2+256^2*k3)-197860+x$ 
```

```
nel caso in cui per esempio M=5000000 si ottiene (passando in hex)
x=0 e k1=0xAC, k2=0x8C, k3=0x03
```

```
*/
```

③ la macchina richiesta può essere ottenuta tramite una architettura di MOORE costituita da un CONTATORE (down) modulo 7 con abilitazione E, seguito da una rete combinatoria RC2 per la produzione dell'uscita.

con $M = 499968$ la sequenza è
0100100

(6543210 stato contatore)

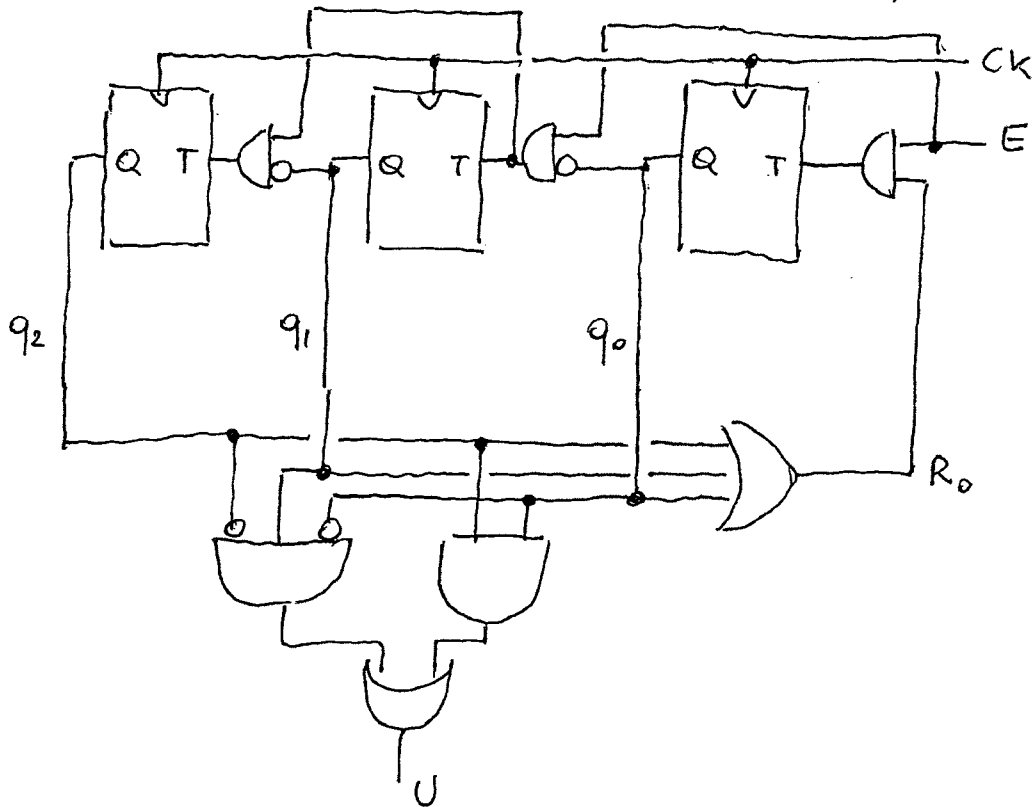
Sintesi di RC2

Scelgo down perché lievemente più semplice

		q_1, q_0			
		00	01	11	10
q_2	0	0 ⁰	0 ¹	0 ³	1 ²
	1	0 ⁴	1 ⁵	-	0 ⁶

$$U = \bar{q}_2 \bar{q}_1 \bar{q}_0 + q_2 q_0$$

Schema logico (per passare da 000 a 110, --B)



4

Funzione combinatoria

13 "1" 11 "0" 8 "-"

2 soli implicati di ordine 3

$$\bar{x}_4 \bar{x}_3 ; \bar{x}_4 \bar{x}_2$$

$x_3 x_2$

$x_1 x_0$	00	01	11	10
00	1	1	1	1
01	1	1	0	1
11	1	1	0	1
10	1	1	0	1

$$x_4 = 0$$

	00	01	11	10
0	-	-	0	-
-	0	-	-	0
0	-	0	-	0
-	0	-	-	0

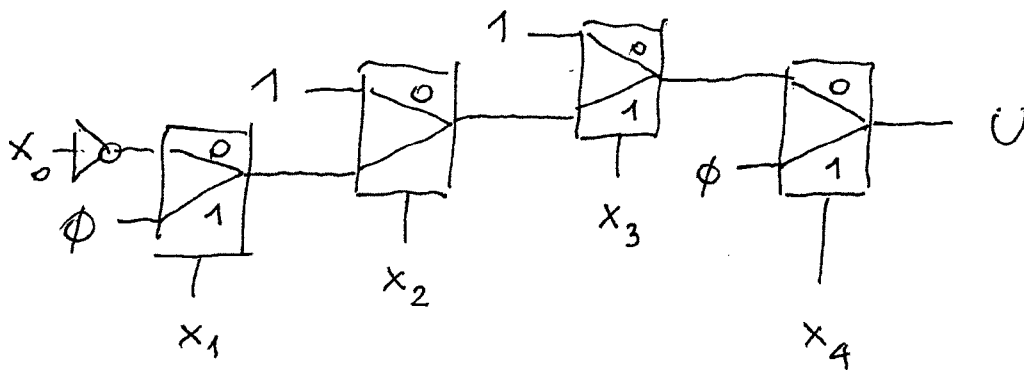
$$x_4 = 1$$

si trova PS ottime, con tutti implicati essenziali

$$\bar{x}_4 (\bar{x}_3 + \bar{x}_2 + \bar{x}_0) (\bar{x}_3 + \bar{x}_2 + \bar{x}_1)$$

- il pallino indica un mintermine coperto in esclusiva dall'implicato essenziale che lo contiene

⑤ Dalla mappa si ricave direttamente:



⑥ Determinare il contenuto della SRAM, separando i campi del microcodice

	Vero	Falso	FL	OPCode
S ₀	110	101	1	00
S ₁	101	011	0	10
S ₂	111	101	0	00
S ₃	101	001	1	10
S ₄	100	001	1	01
S ₅	111	101	1	01
S ₆	100	101	0	01
S ₇	001	011	0	10

flag
 0: G
 1: F

Trascrivo sostituendo

S ₀	S ₆	S ₅	F	OP ₀
S ₁	S ₅	S ₃	G	OP ₂
S ₂	S ₇	S ₅	G	OP ₀
S ₃	S ₅	S ₁	F	OP ₂
S ₄	S ₄	S ₁	F	OP ₁
S ₅	S ₇	S ₅	F	OP ₁
S ₆	S ₄	S ₅	G	OP ₁
S ₇	S ₁	S ₃	G	OP ₂

- Non si può usare un contatore per il calcolo dello stato futuro perché NON ESISTE un percorso CICLICO e COMPLETO (né con flag VERO, né con FALSO; in ogni colonna ci sono stati RIPETUTI)

Diagramma di flusso

