

SCHEDA ASE2407		Data: 15 Luglio 2024
Cognome	Nome	

Il testo deve essere riconsegnato

ESERCIZIO N°1

8 punti

Realizzare un sottoprogramma per il microcontrollore XMEGA256A3BU che ponga in due locazioni di memoria consecutive la differenza (valutata in C2 su 16 bit) tra le somme di gruppi di 32 valori C2 da 1 byte contenuti in locazioni consecutive della memoria. Il byte meno significativo del risultato deve essere posto all'indirizzo puntato da Z. I byte di indirizzo meno significativo di ciascun gruppo da 32 sono puntati rispettivamente da X e da Y. Indicare il range possibile per il risultato.

ESERCIZIO N°2

5 punti

Individuare la rappresentazione binary32 (IEEE-754) della radice quadrata (positiva) dei numeri naturali da 1 a 9 con il minimo errore possibile. Valutare quindi il massimo errore assoluto in modulo commesso e per quale valore si ha tale errore.

ESERCIZIO N°3

5 punti

Sintetizzare un sequenziatore, indicando diagramma di flusso e contenuto della ROM, che fornisca alla parte operativa un numero di 3 bit, secondo la seguente legge:

- 1) se il valore corrente dell'uscita è multiplo di 3 (compreso lo 0) e il flag F è vero, il nuovo valore dell'uscita deve essere uguale al valore attuale incrementato di 3 (modulo 8); se F è falso l'uscita deve essere posta a 1;
- 2) se il valore corrente dell'uscita non è multiplo di 3 e il flag G è vero, il nuovo valore dell'uscita deve essere uguale a 3; se G è falso l'uscita deve essere decrementata di 3 (modulo 8).

La macchina, la cui uscita iniziale è nulla, deve avere un numero di stati tale da poter sempre soddisfare i requisiti indicati. Individuare i valori (eventuali) che non si presentano mai in uscita.

ESERCIZIO N°4

5 punti

Disegnare lo schema logico in forma NAND-NAND ottima di una rete combinatoria non completamente specificata a 5 ingressi, X_4 , X_3 , X_2 , X_1 e X_0 , e 1 uscita Y , i cui maxtermini siano $\{1, 3, 4, 6, 9, 12, 14, 17, 19, 20, 23, 24, 28, 31\}$ e l'insieme dei don't care sia $\{2, 5, 7, 18, 21, 30\}$. Indicare gli implicanti essenziali.

ESERCIZIO N°5

5 punti

Realizzare una macchina sequenziale sincrona secondo il modello di Mealy sincronizzato, con 1 ingresso e 1 uscita che viene posta a 1 (dopo il clock) ogni volta che un ingresso è pari alla somma dei 3 ingressi precedenti (modulo 2). Si trascuri il transitorio iniziale dei primi 3 ingressi.

ESERCIZIO N°6

5 punti

Realizzare la funzione dell'esercizio 4, se possibile, usando al massimo un multiplexer 8:1, 3 multiplexer 4:1 e 3 multiplexer 2:1.

/*Realizzare un sottoprogramma per il microcontrollore XMEGA256A3BU che ponga in due locazioni di memoria consecutive la differenza (valutata in C2 su 16 bit) tra le somme di gruppi di 32 valori C2 da 1 byte contenuti in locazioni consecutive della memoria. Il byte meno significativo del risultato deve essere posto all'indirizzo puntato da Z. I byte di indirizzo meno significativo di ciascun gruppo da 32 sono puntati rispettivamente da X e da Y. Indicare il range possibile per il risultato.*/

```
sub_groups:
    push R2          //registro nullo
    push R16         //contatore
    push R18         //registro ausiliario per i valori da sommare
    push R19         //registro ausiliario per i valori da sottrarre
    push R20         //registri per accumulare il risultato
    push R21
    clr R2
    clr R20
    clr R21
    ldi R16,32
loop:
    ld R18,X+
    ld R19,Y+
    sub R18,R19
    brge oltre      //risultato positivo da sommare
    dec R21         //decrementa la parte alta per tenere conto di S
oltre:
    add R20,R18     //accumula il valore con parte alta nulla
    adc R21,R2
    dec R16
    brne loop
    st Z,R20
    std Z+1,R21     //mette a posto il risultato senza alterare Z
    sbiw XH:XL,32  //ripristina X
    sbiw YH:YL,32  //ripristina Y
    pop R21
    pop R20
    pop R19
    pop R18
    pop R16
    pop R2
ret
```

Il range è: $[-32(128+127);32*(127+128)]$ cioè $[-8160;8160]$ ben rappresentabile con 16 bit in C2 */

2

Individuare la rappresentazione binary32 (IEEE-754) della radice quadrata (positiva) dei numeri naturali da 1 a 9 con il minimo errore possibile. Valutare quindi il massimo errore assoluto in modulo commesso e per quale valore si ha tale errore.

Abbiamo valori che vanno da 1 a 3, coprendo 2 odg binari (esponenti del 2: 0 e 1). Salvo particolarità, ci aspettiamo il massimo errore assoluto con i valori che hanno esponente 1 (5,6,7,8) visto che è maggiore il valore dell'LSB.

n	x=sqrt(n)	e	Eh	T	Th	err(mod)	Codice binario
1	1,00000000	0	7F	0	0000000	0	0_01111111_000.0000.0000.0000.0000.0000
2	1,4142136	0	7F	3474675	3504F3	2,420E-08	0_01111111_011.0101.0000.0100.1111.0011
3	1,7320508	0	7F	6140887	5DB3D7	3,109E-08	0_01111111_101.1101.1011.0011.1101.0111
4	2,00000000	1	80	0	0000000	0	0_10000000_000.0000.0000.0000.0000.0000
5	2,2360680	1	80	990141	0F1BBD	3,283E-08	0_10000000_000.1111.0001.1011.1011.1101
6	2,4494897	1	80	1885297	1CC471	8,914E-08	0_10000000_001.1100.1100.0100.0111.0001
7	2,6457513	1	80	2708477	2953FD	7,320E-08	0_10000000_010.1001.0101.0011.1111.1101
8	2,8284271	1	80	3474675	3504F3	4,841E-08	0_10000000_011.0101.0000.0100.1111.0011
9	3,00000000	1	80	4194304	4000000	0	0_10000000_100.0000.0000.0000.0000.0000

Come si vede in tabella, il massimo errore assoluto in modulo si ha per la radice di 6 e vale: $8,914 \cdot 10^{-8}$

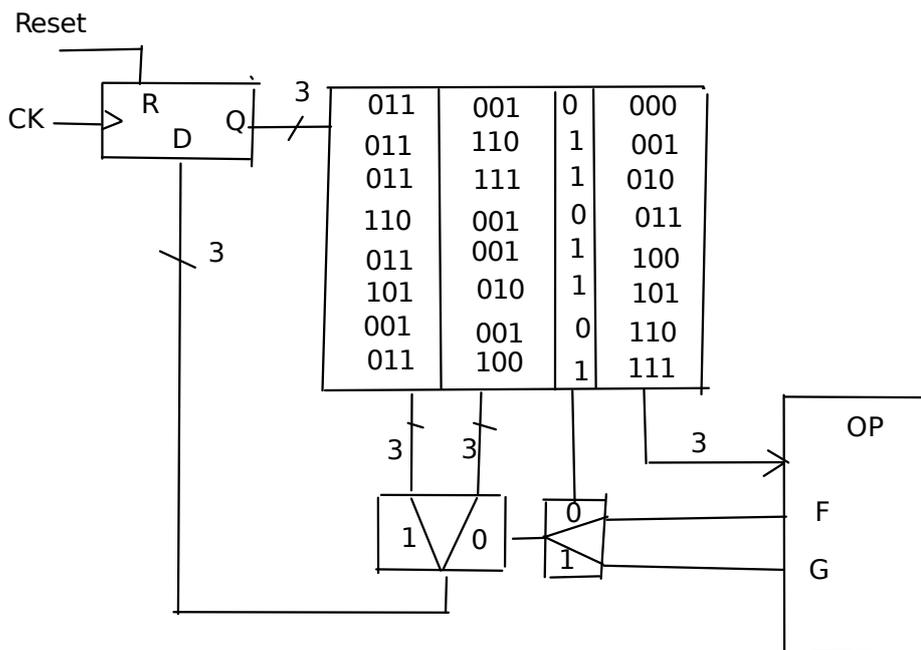
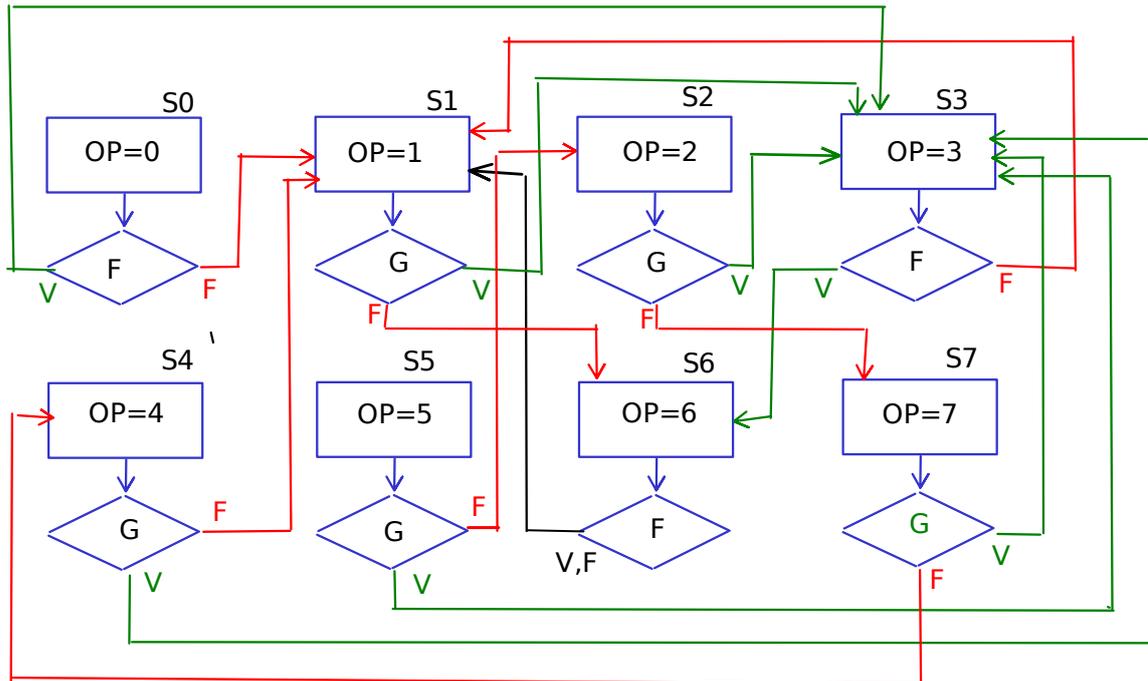
Sintetizzare un sequenziatore, indicando diagramma di flusso e contenuto della ROM, che fornisca alla parte operativa un numero di 3 bit, secondo la seguente legge:

- 1) se il valore corrente dell'uscita è multiplo di 3 (compreso lo 0) e il flag F è vero, il nuovo valore dell'uscita deve essere uguale al valore attuale incrementato di 3 (modulo 8); se F è falso l'uscita deve essere posta a 1;
- 2) se il valore corrente dell'uscita non è multiplo di 3 e il flag G è vero, il nuovo valore dell'uscita deve essere uguale a 3; se G è falso l'uscita deve essere decrementata di 3 (modulo 8).

La macchina, la cui uscita iniziale è nulla, deve avere un numero di stati tale da poter sempre soddisfare i requisiti indicati.

Individuare i valori (eventuali) che non si presentano mai in uscita.

Per il sequenziatore sono sufficienti 8 stati, visto che il valore dell'uscita può assumere al massimo 8 valori (è calcolato modulo 8) e lo stato futuro dipende dall'uscita.

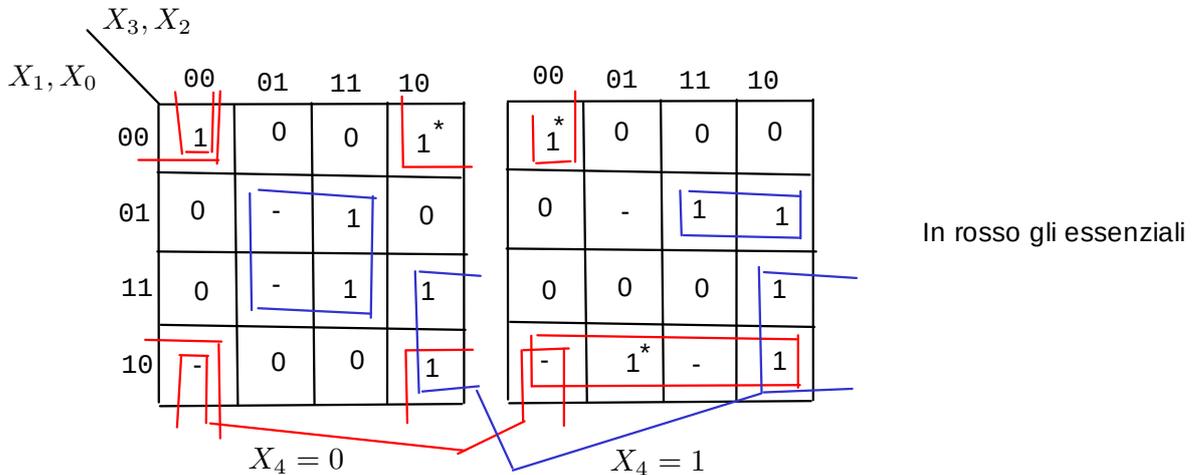


L'uscita 000 si presenta solo all'inizio e non torna più.

Le uscite 101, 010, 111, 100 sono in stati che non possono essere raggiunti. Sulla base di questa osservazione, lo schema può essere semplificato mantenendo solo i 4 stati con uscite 000, 001, 011 e 110. Come codice di questi stati si possono assumere i 2 bit meno significativi.

Disegnare lo schema logico in forma NAND-NAND ottima di una rete combinatoria non completamente specificata a 5 ingressi e 1 uscita, i cui maxtermini siano $\{1, 3, 4, 6, 9, 12, 14, 17, 19, 20, 23, 24, 28, 31\}$ e l'insieme dei don't care sia $\{2, 5, 7, 18, 21, 30\}$. Indicare gli implicanti essenziali.

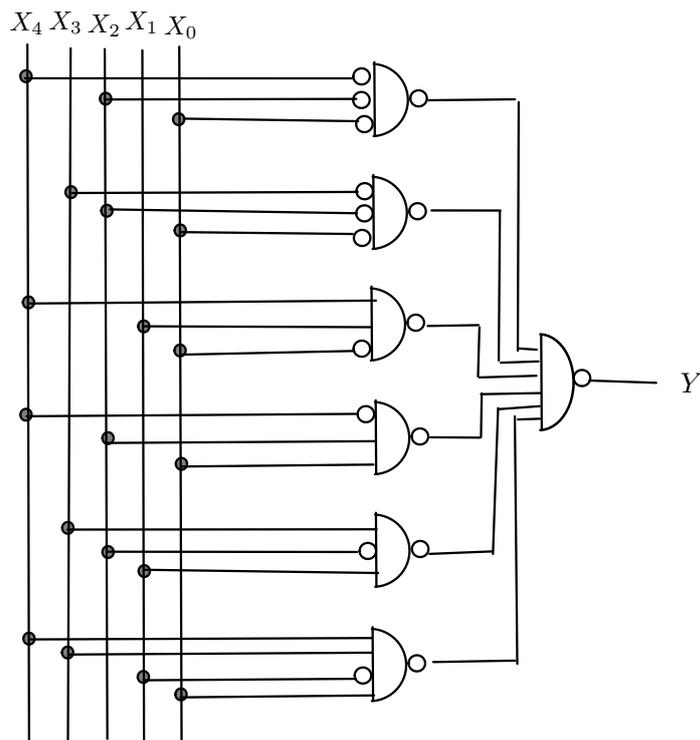
Colloco i maxtermini in mappa e poi eseguo la sintesi cercando implicanti essenziali e principali che coprono la funzione.



Forma SP (6 implicati di cui 3 essenziali, con *)

$$Y = \overline{X_4} \overline{X_2} \overline{X_0} + \overline{X_3} \overline{X_2} \overline{X_0} + X_4 X_1 \overline{X_0} + \overline{X_4} X_2 X_0 + X_3 \overline{X_2} X_1 + X_4 X_3 \overline{X_1} X_0$$

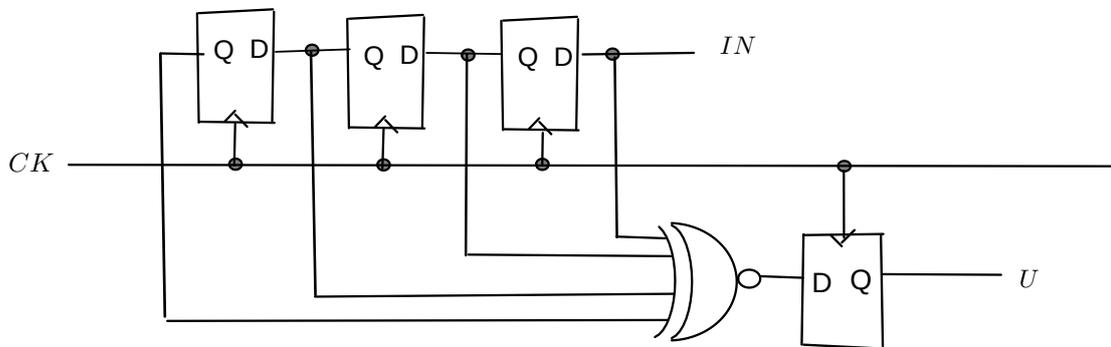
Schema logico a NAND (dopo applicazione t. De Morgan)



5

Realizzare una macchina sequenziale sincrona secondo il modello di Mealy sincronizzato, con 1 ingresso e 1 uscita che viene posta a 1 (dopo il clock) ogni volta che un ingresso è pari alla somma dei 3 ingressi precedenti (modulo 2). Si trascuri il transitorio iniziale dei primi 3 ingressi.

Cerchiamo una soluzione basata su uno shift register a 3 bit, in grado di memorizzare gli ultimi 3 valori arrivati. Per la verifica del requisito si può usare un porta XNOR a 4 ingressi, con l'uscita sincronizzata da un D-FF.

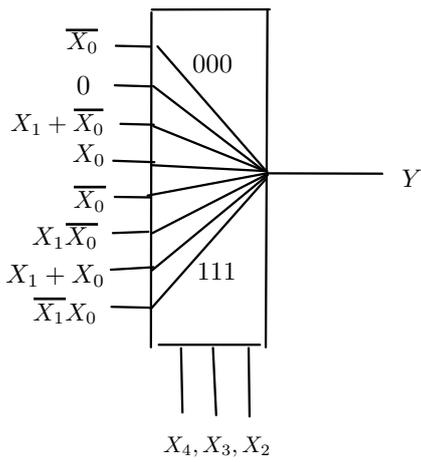


6

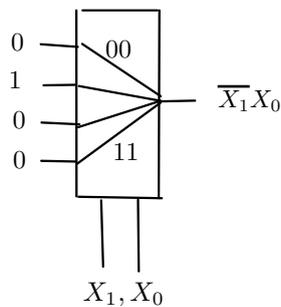
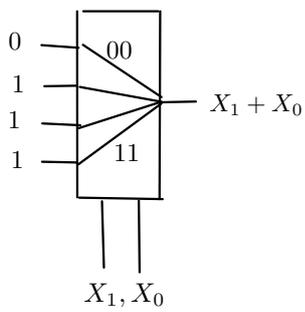
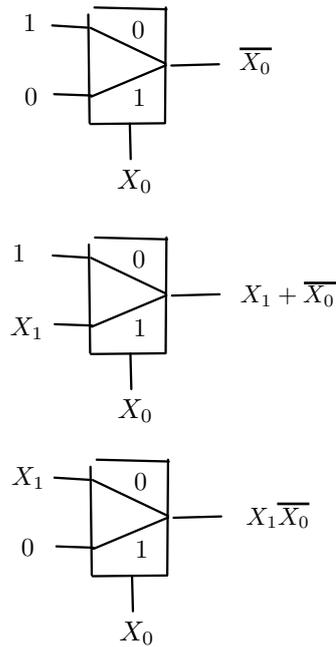
Realizzare la funzione dell'esercizio 4, se possibile, usando al massimo un multiplexer 8:1, 3 multiplexer 4:1 e 3 multiplexer 2:1.

X_3, X_2		X_1, X_0				X_1, X_0			
		00	01	11	10	00	01	11	10
$X_4 = 0$	00	1	0	0	1	1	0	0	0
	01	0	-	1	0	0	-	1	1
	11	0	-	1	1	0	0	0	1
	10	-	0	0	1	-	1	-	1

Usa il mux 8:1 per selezionare le colonne. In ingresso avrò funzioni delle 2 rimanenti variabili.



Usa gli altri mux per ottenere le funzioni necessarie.



Avanza 1 mux 4:1