

Il testo deve essere riconsegnato nella cartellina. **Non usare il colore rosso nello svolgimento.**
 Appello straordinario: l'orale si deve tenere in questa stessa sessione.

ESERCIZIO N°1

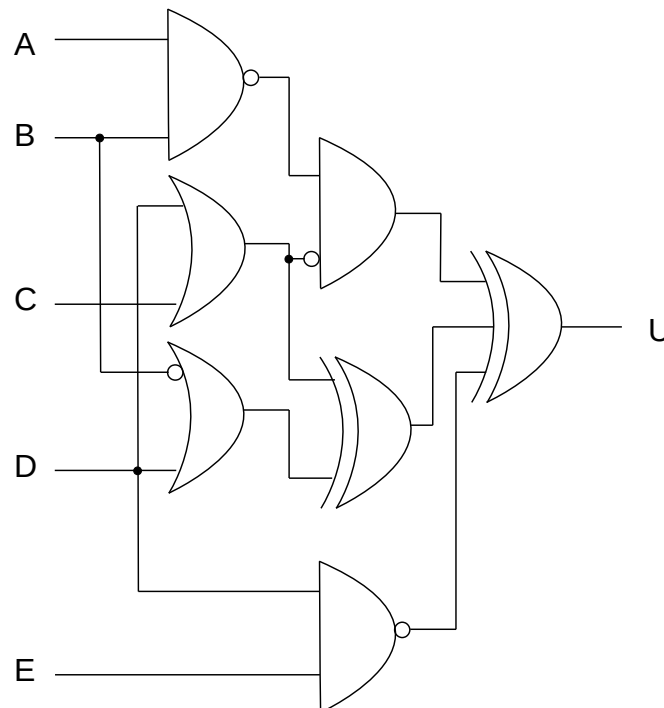
8 punti

Realizzare un sottoprogramma per il microcontrollore XMEGA256A3BU che valuta la parte intera della radice cubica del valore binario senza segno contenuto in X. Il risultato deve essere lasciato nel registro R16.

ESERCIZIO N°2

5 punti

Determinare lo schema logico in forma NOR-NOR minima della rete seguente, evidenziando gli implicati essenziali.



ESERCIZIO N°3

5 punti

Disegnare lo schema logico di un sequenziatore con contatore sincrono (dotato della possibilità di caricamento parallelo) che implementi microcodice specificato nel seguito. Si usi una ROM delle minime dimensioni possibili.

```

A:  IF J THEN F ELSE C; OP = 011100
B:  IF M THEN H ELSE E; OP = 010000
C:  IF K THEN A ELSE D; OP = 101111
D:  IF L THEN C ELSE F; OP = 011100
E:  IF M THEN B ELSE G; OP = 101111
F:  IF J THEN D ELSE B; OP = 101111
G:  IF L THEN E ELSE H; OP = 100011
H:  IF K THEN G ELSE A; OP = 010000
    
```

ESERCIZIO N°4

5 punti

Sintetizzare una rete sequenziale sincronizzata secondo il modello di Mealy sincronizzato, con in ingresso IN e una uscita U , in grado di generare le due sequenze periodiche con il periodo costituito dai valori 1011 quando l'ingresso è 1 e 0010 quando l'ingresso è 0. La macchina è sensibile al valore dell'ingresso solo all'accensione e dopo aver terminato correttamente un ciclo completo della sequenza.

ESERCIZIO N°5

5 punti

Determinare la rappresentazione nel formato frazionale in C2 che, usando il minor numero di bit, approssima il risultato della radice cubica dei numeri interi contenuti in $[-10; 10]$ con un l'errore assoluto massimo (in modulo) minore di 10^{-4} . Valutare tale errore.

ESERCIZIO N°6

5 punti

Disegnare lo schema logico di un registro universale in grado di eseguire, su controllo di un opportuno numero di linee, le istruzioni (con comportamento analogo a quello del microcontrollore) seguenti:

ROR Rd
ROL Rd
ASR Rd
LSR Rd
LSL Rd
LDI Rd, K
ORI Rd, K
ANDI Rd, K

Si includa nello schema anche un flip-flop per il flag C, che deve essere gestito in accordo alle istruzioni proposte.

/*Realizzare un sottoprogramma per il microcontrollore XMEGA256A3BU che valuta la parte intera della radice cubica del valore binario senza segno contenuto in X. Il risultato deve essere lasciato nel registro R16.*/*

cubic_root:

//Il massimo valore ottenibile è 40. Si può partire da 40 e poi scendere

push YL

push YH

ldi R16,40 //per il risultato

loop:

rcall cube //esegue il cubo di R16 e lascia il risultato in R1:R0

cp XL,YL

cpc XH,YH //se Y>X, R16 deve scendere

brcc fine

dec R16

rjmp loop

fine:

pop YH

pop YL

ret

cube:

//esegue il cubo di R16(<41) e lascia il risultato in YH:YL

push R0

push R1

push R2

mul R16,R16 //il quadrato in R1:R0

mov R2,R1 //salva la parte alta in R2

mul R0,R16

movw YH:YL,R1:R0 //salva il parziale nel risultato

mul R2,R16

add YH,R0 //sicuramente rappresentabile, con R1 nullo

pop R2

pop R1

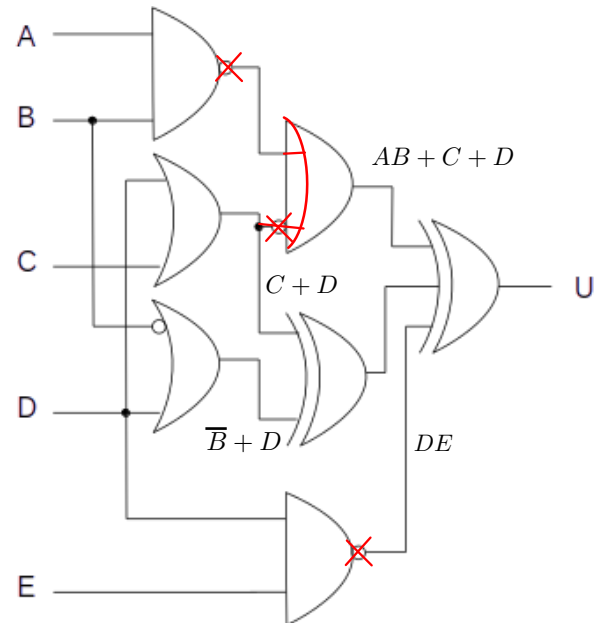
pop R0

ret

2

Disegnare lo schema logico in forma NOR-NOR ottima della rete a fianco. Indicare gli implicati essenziali, giustificando l'affermazione.

Grazie alle proprietà della XOR e al teorema di De Morgan posso applicare allo schema le semplificazioni indicate in rosso. Determino la U con l'espansione di Shannon (variabili D, C e poi B)

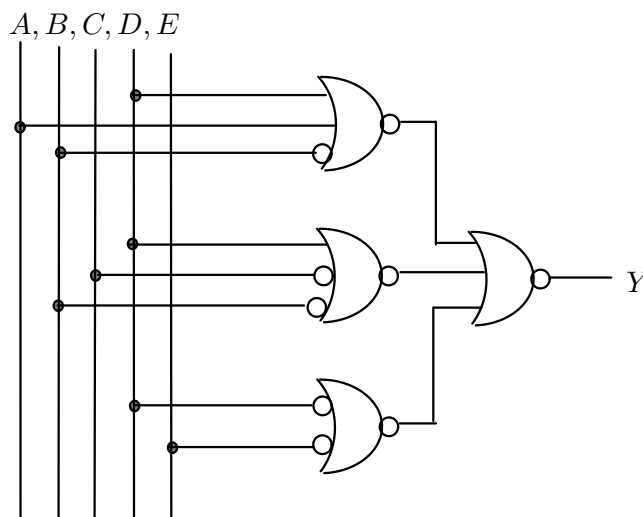


$$\begin{aligned} \overline{D}[\overline{B} \oplus C \oplus (AB + C)] + D\overline{E} &= \\ = \overline{D} \overline{C}(\overline{B} \oplus AB) + \overline{D}C\overline{B} + D\overline{E} &= \\ = \overline{D} \overline{C} \overline{B} + \overline{D} \overline{C}BA + \overline{D}C\overline{B} + D\overline{E} &= U \end{aligned}$$

A, B \ C, E		-							
		00	01	11	10	00	01	11	10
00		1	1	1	1	1	0*	0	1
01		0*	0	0	0	1	0	0	1
11		1	1	0	*0	1	0	0	1
10		1	1	1	1	1	0	0	1
$D = 0$						$D = 1$			

Servono 3 implicati, tutti essenziali. La soluzione ottima ha 8 letterali. Parto dalla forma PS

$$U = (D + A + \overline{B})(D + \overline{C} + \overline{B})(\overline{D} + \overline{E})$$



Disegnare lo schema logico di un sequenziatore con contatore sincrono (dotato della possibilità di caricamento parallelo) che implementi microcodice specificato nel seguito. Si usi una ROM delle minime dimensioni possibili.

```

A:      IF J THEN F ELSE C; OP = 011100
B:      IF M THEN H ELSE E; OP = 010000
C:      IF K THEN A ELSE D; OP = 101111
D:      IF L THEN C ELSE F; OP = 011100
E:      IF M THEN B ELSE G; OP = 101111
F:      IF J THEN D ELSE B; OP = 101111
G:      IF L THEN E ELSE H; OP = 100011
H:      IF K THEN G ELSE A; OP = 010000

```

Cerco una eventuale sequenza ciclica completa.

Vero: A, F, D, C, A (non completa)

Falso: A, C, D, F, B, E, G, H, A (ok, completa)

Riorganizzo le righe secondo la sequenza trovata e codifico gli stati di conseguenza

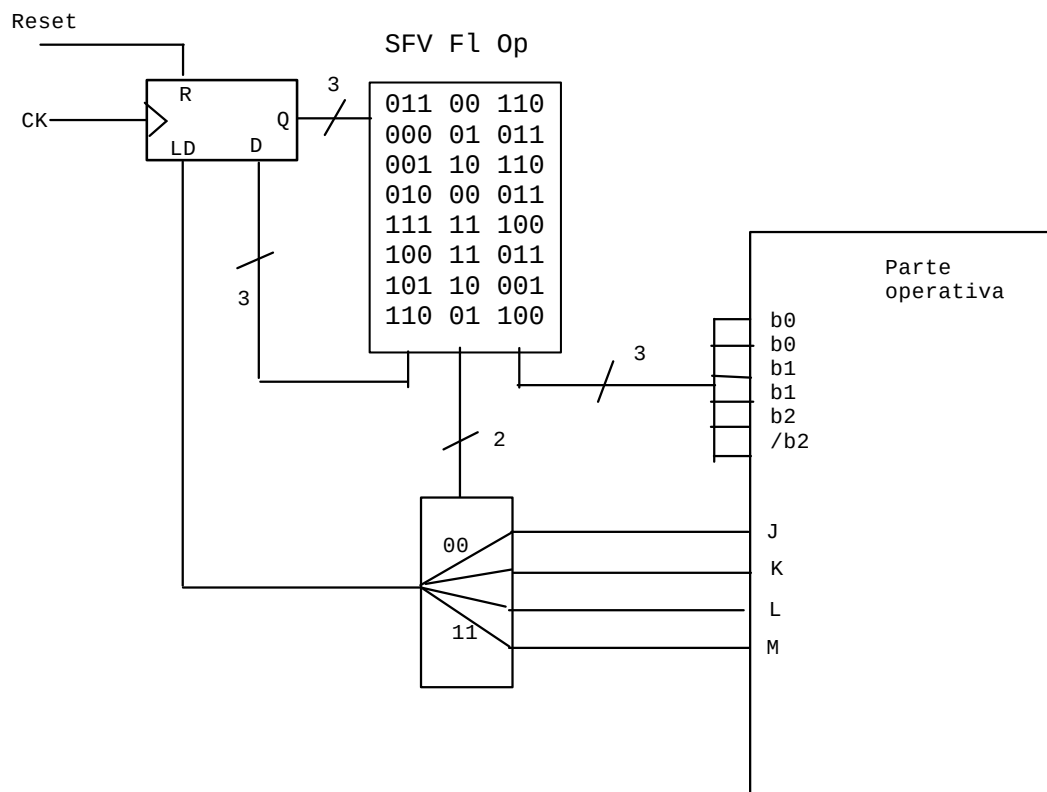
A:	IF J THEN F ELSE C; OP = 011100	A000:F011;C001;J00;Op011100
C:	IF K THEN A ELSE D; OP = 101111	C001:A000;D010;K01;Op101111
D:	IF L THEN C ELSE F; OP = 011100	D010:C001;F011;L10;Op011100
F:	IF J THEN D ELSE B; OP = 101111	F011:D010;B100;J00;Op101111
B:	IF M THEN H ELSE E; OP = 010000	B100:H111;E101;M11;Op010000
E:	IF M THEN B ELSE G; OP = 101111	E101:B100;G110;M11;Op101111
G:	IF L THEN E ELSE H; OP = 100011	G110:E101;H111;L10;Op100011
H:	IF K THEN G ELSE A; OP = 010000	H111:G110;A000;K01;Op010000

Lo stato futuro se falso può essere determinato dal contatore con incremento.

Come ultima semplificazione, osservo che i 6 bit dell'istruzione possono ridursi a 3.

Infatti la prima coppia è sempre costituita da bit opposti mentre le altre 2 da bit uguali.

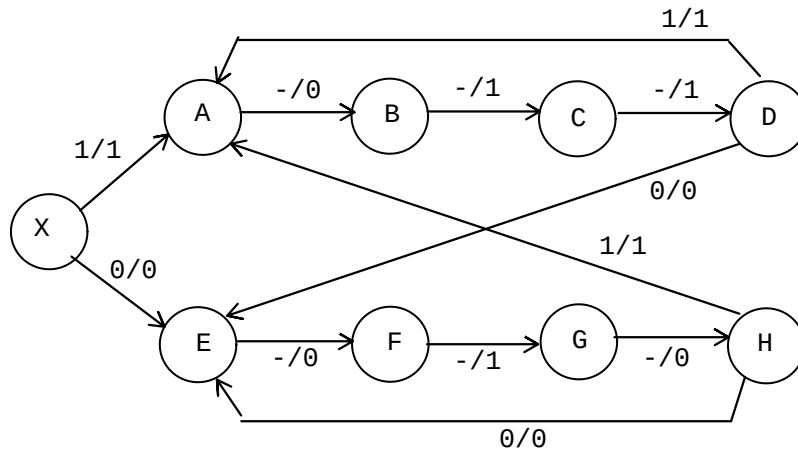
La ROM necessaria ha quindi 8 parole da 8 bit.



4

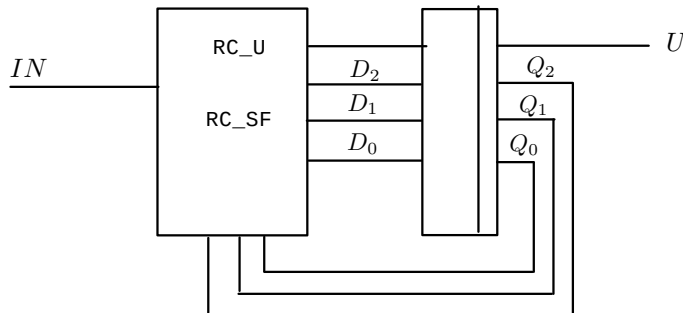
Sintetizzare una rete sequenziale sincronizzata secondo Mealy sincronizzato, con in ingresso IN e una uscita U, in grado di generare le due sequenze periodiche con il periodo costituito dai valori 1011 quando l'ingresso è 1 e 0010 quando l'ingresso è 0.

La macchina è sensibile al valore dell'ingresso solo all'accensione e dopo aver terminato correttamente un ciclo completo della sequenza.



Lo stato iniziale X e quelli finali D e H sono equivalenti, in quanto mostrano una identica evoluzione a parità di ingressi.

Architettura



XDH 000
A 001
B 011
C 010
E 101
F 111
G 110

Mappe delle transizioni e dell'uscita

Q_1, Q_0		Q_2, IN			
		00	01	11	10
00	101/0	011/0	010/1	000/1	
01	001/1	011/0	010/1	000/1	
11	---/-	111/0	110/1	000/0	
10	---/-	111/0	110/1	000/0	

$$D_2 = \overline{IN} \overline{Q_1} \overline{Q_0} + Q_2 Q_0 \quad D_1 = Q_0 \quad D_0 = \overline{Q_1}$$

0	0	1	1
1	0	1	1
-	0	1	0
-	0	1	0

$$U = \overline{IN} \overline{Q_1} \overline{Q_0} + Q_1 Q_0 + \overline{Q_2} Q_1$$

Determinare la rappresentazione nel formato frazionale in C2 che, usando il minor numero di bit, approssima il risultato della radice cubica dei numeri interi contenuti in $[-10; 10]$ con un l'errore assoluto massimo (in modulo) minore di 0,1 millesimi. Valutare tale errore.

Procedura: sicuramente, visto il range del risultato e l'errore richiesto, una notazione [3.13] soddisfa la richiesta fatta (si ricorda che l'errore di arrotondamento è pari a $\text{LSB}/2$).

Si possono calcolare quindi i codici e l'errore massimo.

Infine, per confermare che quella trovata è la notazione con il minor numero di bit che soddisfa il requisito, è sufficiente trovare almeno un caso in cui l'uso di un bit in meno [3.12] porta a un errore assoluto in modulo maggiore di 0,1 millesimi.

$$\text{range} \quad -4 \leq \hat{y} \leq 4 - 2^{-13}$$

$$\text{errore } |\epsilon| \leq 2^{-14} \simeq 6,104 \cdot 10^{-5}$$

Osservazione: si possono prendere in considerazione solo i valori non negativi (l'errore di arrotondamento di quelli negativi sarà il medesimo), scartando 0, 1 e 8 che danno valori esatti, con errore nullo. È sufficiente considerare 2, 3, 4, 5, 6, 7, 9 e 10

$$\epsilon_n = \frac{|\text{round}(2^{13} \sqrt[3]{n}) - 2^{13} \sqrt[3]{n}|}{2^{13}}$$

$$\epsilon_n = \frac{|\text{round}(2^{12} \sqrt[3]{n}) - 2^{12} \sqrt[3]{n}|}{2^{12}}$$

Valutazione della formula
per [3.13] bit

2	3,34E-05
3	1,12E-05
4	1,29E-06
5	1,50E-05
6	1,81E-05
7	3,27E-05
9	5,70E-06
10	1,57E-05

Valutazione della formula
per [3.12]

2	8,87E-05
3	1,11E-04
4	1,29E-06
5	1,50E-05
6	1,81E-05
7	8,94E-05
9	5,70E-06
10	1,06E-04

Conclusione:

dalla formula calcolata per [3.13] si vede che si ha il massimo errore assoluto in modulo per 2 (e -2) e vale 0,0334 millesimi

se si riduce la rappresentazione di 1 bit nella parte frazionale (la parte intera non può essere toccata, perché si avrebbe un errore molto maggiore per i valori superiori a 8) e quindi [3.12], il massimo errore assoluto in modulo è maggiore di 0,1 millesimi per 3 e 10.

6

Disegnare lo schema logico di un registro universale in grado di eseguire, su controllo di un opportuno numero di linee, le istruzioni indicate (con comportamento analogo a quello del microcontrollore)

S: Operation

- 0: ROR Rd
- 1: ROL Rd
- 2: ASR Rd
- 3: LSR Rd
- 4: LSL Rd
- 5: LDI Rd, K
- 6: ORI Rd, K
- 7: ANDI Rd, K

