

Il testo deve essere riconsegnato nella cartellina. **Non usare il colore rosso nello svolgimento.**

ESERCIZIO N°1

8 punti

Realizzare un sottoprogramma per il microcontrollore AVR XMEGA256A3BU, in grado di moltiplicare il numero con segno (C2) da 16 bit, contenuto all'indirizzo puntato da X, con il numero senza segno da 16 bit, contenuto all'indirizzo puntato da Y. Il risultato (valutare quanti byte sono necessari per la sua rappresentazione senza errori in C2) deve essere posto in memoria all'indirizzo puntato da Z. Per i valori numerici multibyte, le parti più significative hanno (come d'uso) indirizzi maggiori.

ESERCIZIO N°2

5 punti

Realizzare in forma NAND-NAND ottima una rete combinatoria in grado di valutare la seguente espressione, il cui risultato è booleano (x_1, x_0 e y_2, y_1, y_0 sono le cifre dei 2 numeri binari X e Y). Evidenziare i termini essenziali.

$$|7XY|_5 \geq |X^3 + Y^2|_7$$

ESERCIZIO N°3

5 punti

Realizzare un generatore di forma d'onda con un ingresso X in grado di produrre le due forme d'onda periodiche con periodo 0110110 se l'ingresso X è 0 oppure 01010 se l'ingresso X è 1. La variabile di ingresso viene osservata in occasione del primo fronte di clock dopo l'accensione e poi solo al termine di ciascun periodo. Si usi una architettura di Moore.

ESERCIZIO N°4

5 punti

Realizzare la seguente funzione booleana

$$u = \bar{x} + y\bar{z} + z(x\bar{t} + t\bar{y})$$

a) usando dei multiplexer 2:1

b) usando un decoder 4:16 e il minimo numero possibile di porte NOR a 3 ingressi

ESERCIZIO N°5

5 punti

Determinare la codifica in binary32 dei seguenti valori (scegliendo il codice più vicino):

$$-e^{e/2}$$

$$\log_3(0,145)$$

$$-1/100!$$

ESERCIZIO N°6

5 punti

Disegnare lo schema logico di un sommatore a 3 bit secondo l'approccio carry look-ahead. Si hanno a disposizione porte elementari: AND, OR, NOT, XOR.

```

/*Realizzare un sottoprogramma per il microcontrollore AVR XMEGA256A3BU,
in grado di moltiplicare il numero con segno (C2) da 16 bit,
contenuto all'indirizzo puntato da X, con il numero senza segno da 16 bit,
contenuto all'indirizzo puntato da Y.
Il risultato (valutare quanti byte sono necessari per la sua rappresentazione
senza errori in C2) deve essere posto in memoria all'indirizzo puntato da Z.
Per i valori numerici multibyte, le parti più significative hanno (come d'uso)
indirizzi maggiori. */

```

```

/*Il risultato sarà compreso tra  $-2^{15} \cdot (2^{16}-1)$  e  $(2^{15}-1)(2^{16}-1)$  quindi
tra  $-2^{31}+2^{15}$  e  $2^{31}-3 \cdot 2^{15}+1$  quindi 4 Byte sono sufficienti per
la rappresentazione senza errori in C2 che permette di rappresentare valori
tra  $-2^{31}$  e  $2^{31}-1$  */

```

```

mulsu_16x16:

```

```

push R0
push R1
push R2
push R16 //appoggio operando signed
push R17
push R18 //appoggio operando unsigned
push R19
push R20 //appoggio risultato
push R21
push R22
push R23
clr R2 //registro per estensione del segno
ld R16,X+ //valore unsigned
ld R17,X //valore signed
ld R18,Y //valore unsigned
ldd R19,Y+1 //valore unsegno
mul R16,R18 //unsigned x unsigned
movw R21:R20,R1:R0
mulsu R17,R19 //signed x unsigned (parte più significativa)
movw R23:R23,R1:R0
mul R16,R19 //unsigned x unsigned (parte intermedia)
add R21,R0
adc R22,R1
adc R23,R2 //estensione con 0
mulsu R17,R18 //signed x unsigned (parte intermedia)
sbc R2,R2 //estensione in C2
add R21,R0
adc R22,R1
adc R23,R2
st Z,R20
std Z+1,R21
std Z+2,R22
std Z+3,R23
sbiw XH:XL,1
pop R23
pop R22
pop R21
pop R20
pop R19
pop R18
pop R17
pop R16
pop R2
pop R1
pop R0
ret

```

2

Realizzare in forma NAND-NAND ottima una rete combinatoria in grado di valutare l'espressione nel seguito, il cui risultato è booleano. (x1, x0 e y2, y1, y0 sono le cifre dei 2 numeri binari X e Y). Evidenziare i termini essenziali.

$$|7XY|_5 \geq |X^3 + Y^2|_7$$

$$\begin{aligned} X = 0 &\implies 0 \geq |Y^2|_7 \\ X = 1 &\implies |2Y|_5 \geq |1 + Y^2|_7 \\ X = 2 &\implies |4Y|_5 \geq |1 + Y^2|_7 \\ X = 3 &\implies |Y|_5 \geq |6 + Y^2|_7 \end{aligned}$$

x_1, x_0		y_1, y_0				$y_2 = 0$				$y_2 = 1$			
		00	01	11	10	00	01	11	10	00	01	11	10
0	00	*1	0	0	0	0	1	0	0	0	1	0	
1	01	0	*1	0	0	*1	0	1	1	0	1	1	
3	11	0	1	1*	0	1	0	0	1	0	0	1	
2	10	0	*1	0	0	0	0	1	0	0	1	1	
		0	1	3	2	4	5	7	6				

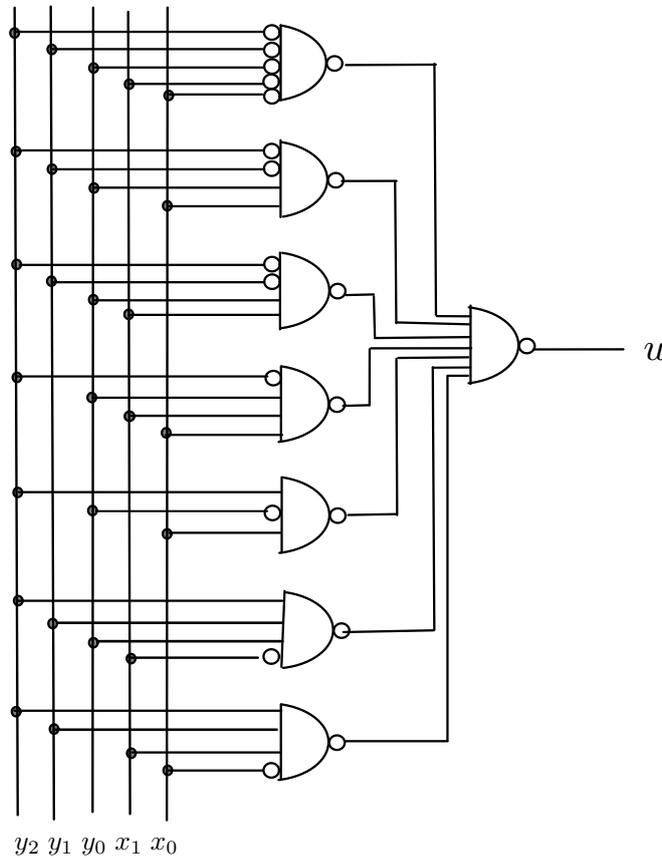
Parto da sintesi ottima SP

7 implicanti di cui
5 essenziali

28 letterali

$$\begin{aligned} u = & \overline{y_2} \overline{y_1} \overline{y_0} \overline{x_1} \overline{x_0} + \overline{y_2} \overline{y_1} y_0 x_0 + \overline{y_2} \overline{y_1} y_0 x_1 + \overline{y_2} y_0 x_1 x_0 + \\ & + y_2 \overline{y_0} x_0 + y_2 y_1 y_0 \overline{x_1} + y_2 y_1 x_1 \overline{x_0} \end{aligned}$$

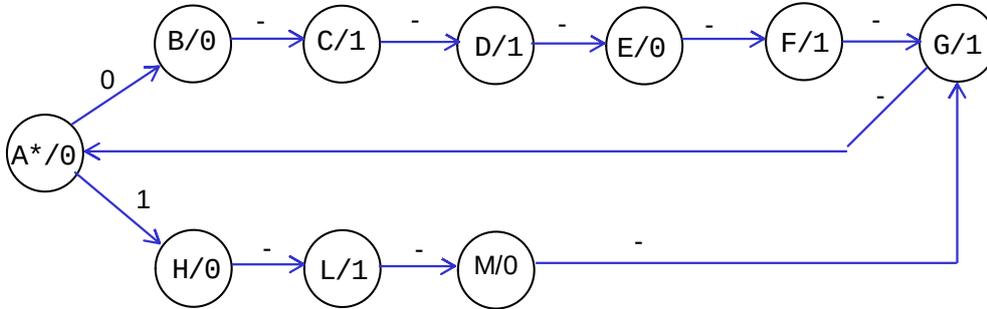
Schema NAND-NAND



3

Realizzare un generatore di forma d'onda con un ingresso X in grado di produrre le due forme d'onda periodiche con periodo 0110110 se l'ingresso X è 0 oppure 01010 se l'ingresso X è 1. La variabile di ingresso viene osservata in occasione del primo fronte di clock dopo l'accensione e poi solo al termine di ciascun periodo.

Grafo delle transizioni



10 stati, servono almeno 4 variabili di stato

Mappa delle transizioni

Q_3, Q_2		Q_1, Q_0				Q_3, Q_2			
		00	01	11	10	00	01	11	10
Q_1, Q_0	00	0001	----	----	----	0010	----	----	----
	01	1001	----	0011	1101	1001	----	0011	1101
	11	1011	----	0000	1111	1011	----	0000	1111
	10	1010	1111	----	0110	1010	1111	----	0110

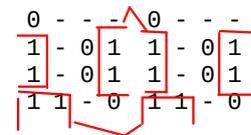
$IN = 0$ $IN = 1$

Codifica degli stati

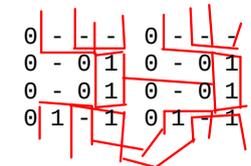
- A 0000 (0)
- B 0001 (1)
- C 1001 (9)
- D 1101 (D)
- E 0011 (3)
- F 1011 (B)
- G 1111 (F)
- H 0010 (2)
- L 1010 (A)
- M 0110 (6)

Rete per l'uscita: cortocircuito

$$U = Q_3$$



$$D_3 = \overline{Q_2}Q_0 + \overline{Q_3}Q_1\overline{Q_0}$$



$$D_2 = Q_2\overline{Q_0} + Q_3\overline{Q_2}Q_0 + Q_3\overline{Q_0}$$

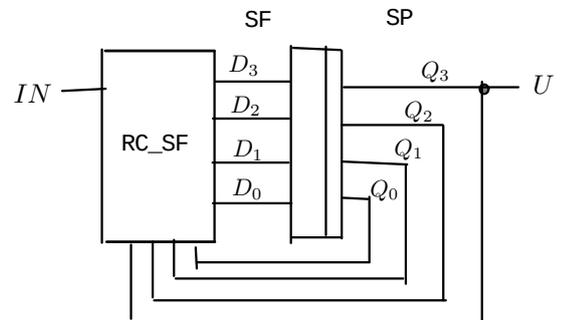


$$D_1 = IN\overline{Q_0} + \overline{Q_3}Q_2 + Q_2\overline{Q_1} + \overline{Q_2}Q_1$$



$$D_0 = \overline{IN}Q_3\overline{Q_1} + \overline{Q_3}Q_2 + Q_2\overline{Q_1} + \overline{Q_2}Q_0$$

Architettura



Realizzare la seguente funzione booleana

a) usando dei multiplexer 2:1

b) usando un decoder 4:16 e il minimo numero possibile di porte NOR a 3 ingressi

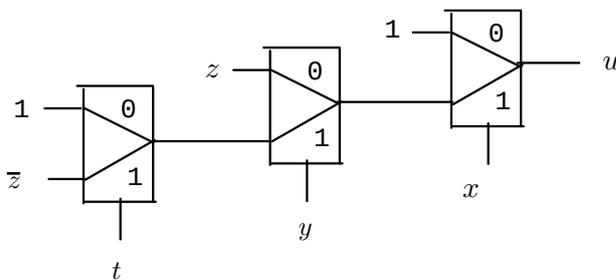
$$u = \bar{x} + y\bar{z} + z(x\bar{t} + t\bar{y})$$

Passiamo in forma SP e determiniamo la mappa:

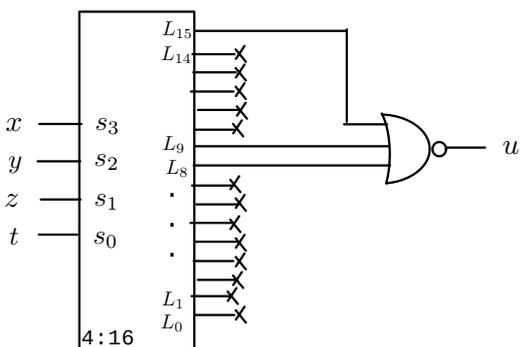
$$u = \bar{x} + y\bar{z} + zx\bar{t} + zt\bar{y}$$

x, y		z, t			
		00	01	11	10
z, t	00	1	1	1	0
	01	1	1	1	0
	11	1	1	0	1
	10	1	1	1	1

Soluzione con mux 2:1



Soluzione con decoder e NOR



5

Determinare la codifica in binary32 dei seguenti valori (scegliendo il codice più vicino):

$$A = -e^{e/2}$$

$$B = \log_3(0,145)$$

$$C = 1/100!$$

Valutiamo il valore approssimato dei tre valori

(con tutta la precisione della calcolatrice, quindi molto maggiore del binary32).

Esprimiamo quindi S, E e T in decimale ed esadecimale. Troviamo infine il codice binario.

$$A \simeq -3,89284757491$$

$$S=1$$

$$E=128=0x80$$

$$T=7939178=0x79246A$$

$$\text{code: } 1 \ 10000000 \ 111 \ 1001 \ 0010 \ 0100 \ 0110 \ 1010$$

$$B \simeq -1,75769155$$

$$S=1$$

$$E=127=0x7F$$

$$T=6355977=0x60FC09$$

$$\text{code: } 1 \ 01111111 \ 110 \ 0000 \ 1111 \ 1100 \ 0000 \ 1001$$

$$C \simeq -1,07 \cdot 10^{-158} \quad -2^{-149} \ll C < 0$$

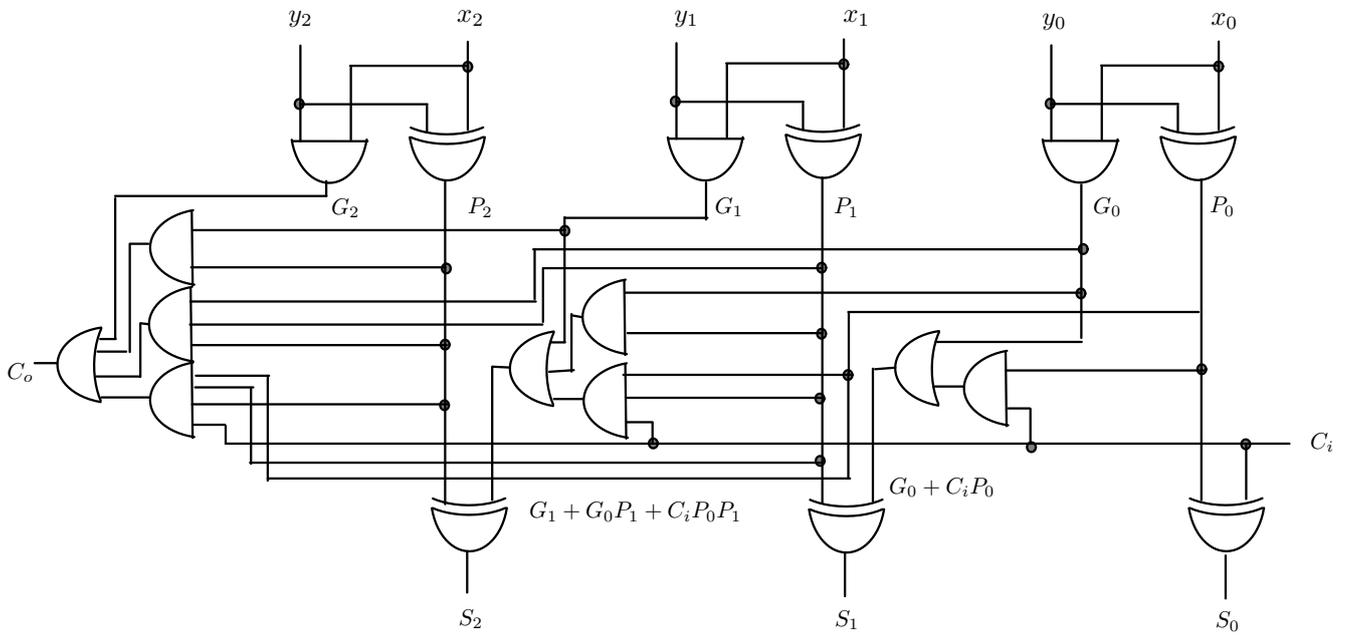
$$S=1$$

$$E=0$$

$$T=0$$

$$\text{code: } 1 \ 00000000 \ 000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000$$

- 6 Disegnare lo schema logico di un sommatore a 3 bit secondo l'approccio carry look-ahead. Si hanno a disposizione le porte elementari.



$$C_o = G_2 + G_1 P_2 + G_0 P_1 P_2 + C_i P_0 P_1 P_2$$