

<b>SCHEDA ASE1706</b>		Data: <b>17 Luglio 2017</b>
Cognome	Nome	

### ESERCIZIO N°1

8 punti

Realizzare un sottoprogramma per il microcontrollore AVR XMEGA256A3BU, in grado di valutare l'ordinata di una retta  $y = ax + b$ . I coefficienti  $a$  e  $b$  sono valori da 1 byte in C2 contenuti in due locazioni successive della memoria a partire dall'indirizzo contenuto in Z, mentre l'ascissa  $x$ , anch'essa su un byte in C2, è contenuta in R16.

Per il risultato, da rappresentare in C2 su un numero adeguato di byte (che deve essere valutato), sono disponibili i registri a partire da R18.

### ESERCIZIO N°2

4 punti

Disegnare lo schema logico di un sequenziatore in grado di implementare il seguente microcodice. R è lo stato iniziale e va codificato con 0.

```
A:  IF L THEN R ELSE A; OP = 101
B:  IF L THEN R ELSE B; OP = 100
C:  IF M THEN G ELSE B; OP = 110
D:  IF M THEN F ELSE F; OP = 011
E:  IF L THEN E ELSE C; OP = 111
F:  IF M THEN D ELSE R; OP = 010
G:  IF L THEN A ELSE D; OP = 000
R:  IF M THEN B ELSE E; OP = 001
```

### ESERCIZIO N°3

6 punti

Disegnare il grafo di una macchina sequenziale sincrona secondo il modello di Moore con un ingresso e una uscita in grado di riconoscere - ponendo 1 in uscita per 1 ciclo di clock - tutte le sequenze non interallacciate di 3 valori che hanno più 1 che 0. Si preveda l'inizializzazione della macchina con l'uso di flip-flop con reset asincrono, in modo che non avvengano riconoscimenti spuri nella fase iniziale. Minimizzare il numero di stati. Presentare lo schema a blocchi dell'architettura corrispondente. Non è richiesta la sintesi.

### ESERCIZIO N°4

5 punti

Determinare, tra tutte le forme normali SP, quella a minimo numero di letterali che realizza una funzione logica in grado di evidenziare ponendo 1 in uscita se il numero binario di ingresso a 5 bit sia pari oppure multiplo di 3, o di 5, o di 7. Evidenziare gli implicanti essenziali. Disegnare quindi lo schema logico corrispondente.

## **ESERCIZIO N°5**

4 punti

Disegnare lo schema logico di un contatore Johnson modulo 7. Prevedere, disponendo di D-FF con reset asincrono, la possibilità di inizializzare in modo asincrono il contatore al valore 1000000.

## **ESERCIZIO N°6**

6 punti

Siano dati i 2 valori razionali  $17/11$  e  $178/129$ .

- a) Determinare la rappresentazione arrotondata in notazione 1.7 e valutare l'errore di rappresentazione (assoluto e relativo) commesso nei 2 casi.
- b) Valutare la rappresentazione del risultato del prodotto tra i due numeri così come ottenuta usando l'operazione FMUL di un microcontrollore AVR [R1:R0]. Indicare anche il valore del flag C.

# 1

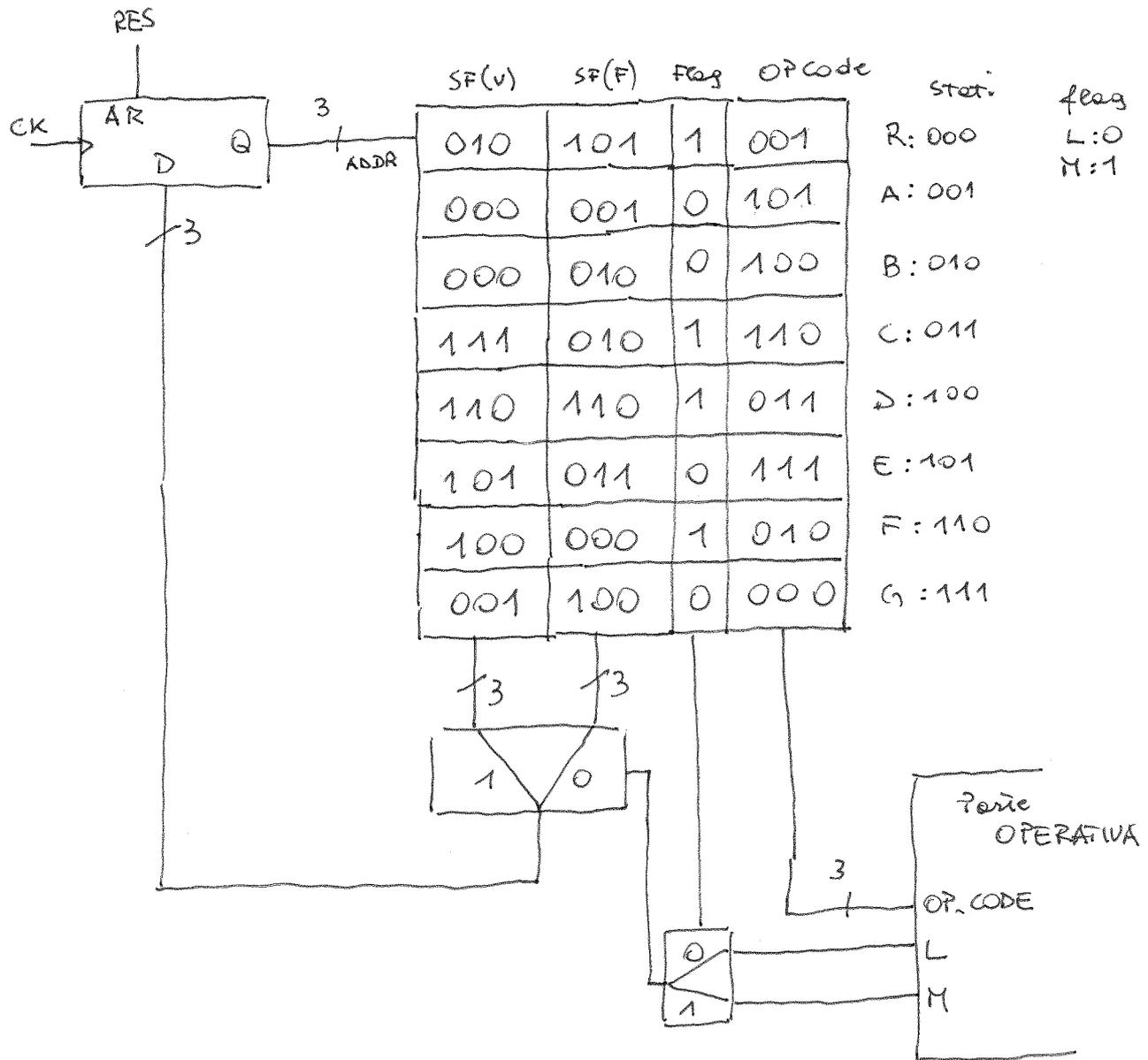
Realizzare un sottoprogramma per il microcontrollore AVR XMEGA256A3BU, in grado di valutare l'ordinata di una retta  $y = ax + b$ . I coefficienti  $a$  e  $b$  sono valori da 1 byte in C2 contenuti in due locazioni successive della memoria a partire dall'indirizzo contenuto in Z, mentre l'ascissa  $x$ , anch'essa su un byte in C2, è contenuta in R16.

Per il risultato, da rappresentare in C2 su un numero adeguato di byte (che deve essere valutato), sono disponibili i registri a partire da R18.

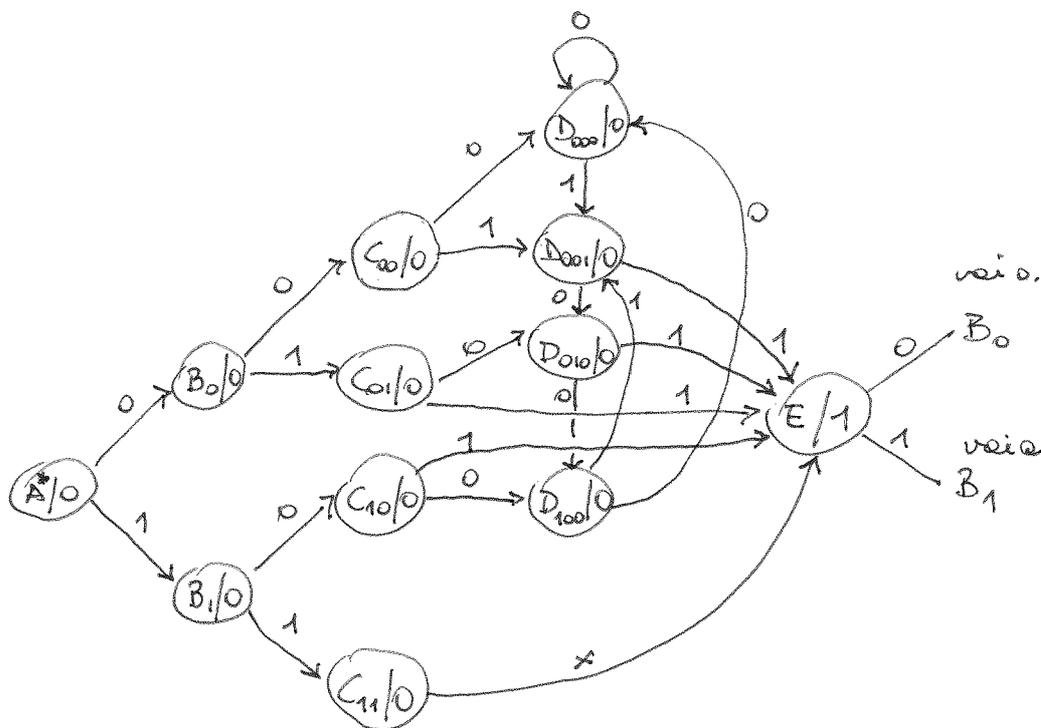
```
/* Il risultato è rappresentabile in C2 su 16 bit.
 * Il min è  $-128*127-128 = -2^{14} > -2^{15}$ .
 * Il max è  $-128*(-128)+127 = 2^{14}+127 < 2^{15}-1$ .
 * Si userà R19:R18
 */
```

```
eval_y:
push R0 //salva registri usati
push R1
ld R18,Z //a in R18
muls R16,R18 //ax in R1:R0
ldd R18,Z+1 //b in R18
ldi R19,-1 //inizializza R19 per R18 negativo
tst R18
brmi skip
clr R19 //azzera se R18 positivo
skip:
add R18,R0
adc R19,R1 //ax+b in R19:R18
pop R1 //ripristina i registri
pop R0
ret
```

② Sequenziatore con registro (R stato iniziale)



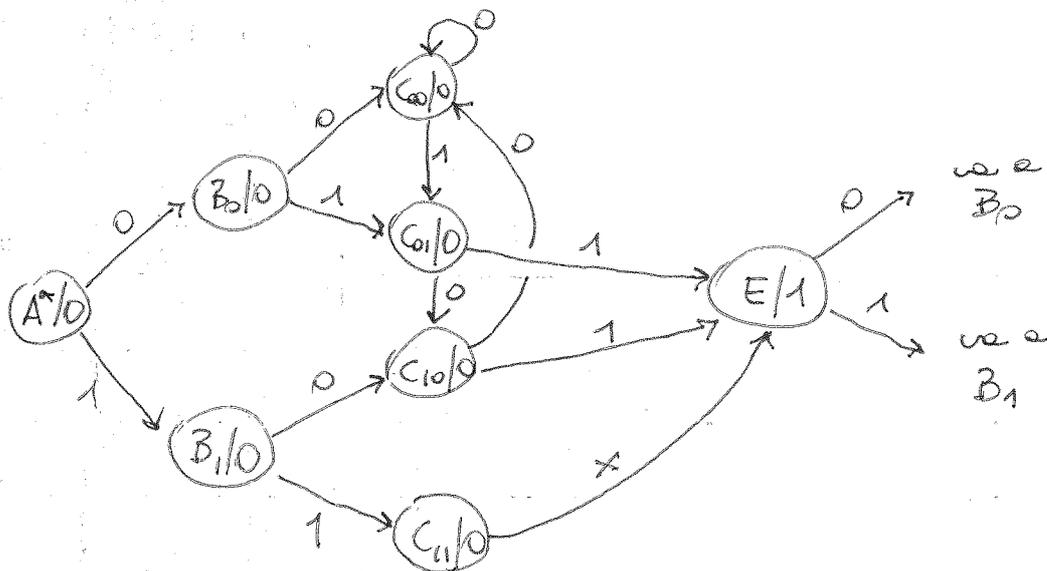
- ③ Partiamo da un grafo che tiene conto di tutte le alternative di ingresso.  
 Poiché non si vuole intersecciamiento, tutti gli stati di riconoscimento ( $v=1$ ) sono equivalenti e poi occorre ripartire da capo



La macchina ha 12 stati. Esaminiamo le eventuali equivalenze.

Si vede facilmente che  $C_{00}$ ,  $D_{000}$  e  $D_{100}$  hanno gli stessi stati futuri. Come pure  $C_{01}$  e  $D_{001}$ . E anche  $C_{10}$  e  $D_{010}$ .

Disegnare il nuovo grafo, che ha 8 stati.



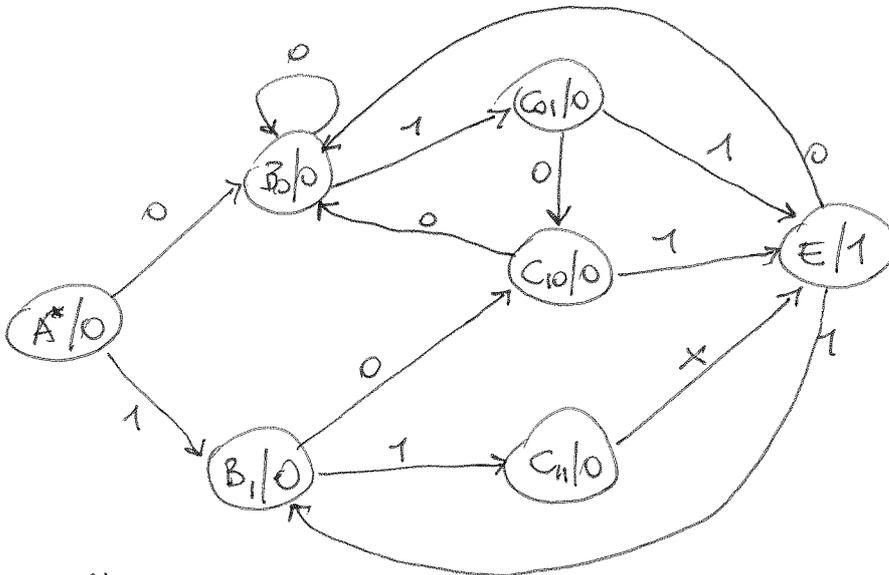
Esaminiamo le possibilità di ulteriori equivalenze

A	OK							
B <sub>0</sub>	$\begin{matrix} C_{00} \times B_0 \\ C_{01} \times B_1 \end{matrix}$	OK						
B <sub>1</sub>	$\begin{matrix} C_{10} \times B_0 \\ C_{11} \times B_1 \end{matrix}$	$\begin{matrix} C_{10} \times C_{00} \\ C_{11} \times C_{01} \end{matrix}$	OK					
C <sub>00</sub>	$\begin{matrix} C_{00} \times B_0 \\ C_{01} \times B_1 \end{matrix}$	$C_{00} = C_{00}$	$\begin{matrix} C_{00} \times C_{10} \\ C_{01} \times C_{11} \end{matrix}$	OK				
C <sub>01</sub>	$\begin{matrix} C_{10} \times B_0 \\ E \times B_1 \end{matrix}$	$\begin{matrix} C_{10} \times C_{00} \\ E \times C_{01} \end{matrix}$	$\begin{matrix} C_{10} \times C_{10} \\ E \times C_{11} \end{matrix}$	$\begin{matrix} C_{10} \times C_{00} \\ E \times C_{01} \end{matrix}$	OK			
C <sub>10</sub>	$\begin{matrix} C_{00} \times B_0 \\ E \times B_1 \end{matrix}$	$\begin{matrix} C_{00} \times C_{00} \\ E \times C_{01} \end{matrix}$	$\begin{matrix} C_{00} \times C_{10} \\ E \times C_{11} \end{matrix}$	$\begin{matrix} C_{00} \times C_{00} \\ E \times C_{01} \end{matrix}$	$\begin{matrix} C_{00} \times C_{10} \\ E \times E \end{matrix}$	OK		
C <sub>11</sub>	$\begin{matrix} E \times B_0 \\ E \times B_1 \end{matrix}$	$\begin{matrix} E \times C_{00} \\ E \times C_{01} \end{matrix}$	$\begin{matrix} E \times C_{10} \\ E \times C_{11} \end{matrix}$	$\begin{matrix} E \times C_{00} \\ E \times C_{01} \end{matrix}$	$\begin{matrix} E \times C_{10} \\ E \times E \end{matrix}$	$\begin{matrix} E \times C_{00} \\ E \times E \end{matrix}$	OK	
E	X	X	X	X	X	X	X	OK
	A	B <sub>0</sub>	B <sub>1</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>10</sub>	C <sub>11</sub>	E (uscita 1)

Iterando diverse volte si escludono molte possibili equivalenze.

Si ottiene  $C_{00} \equiv B_0$

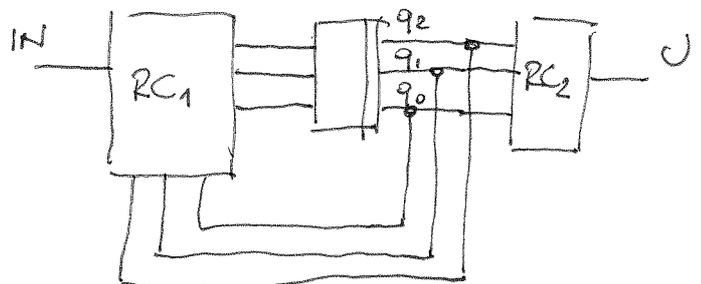
grafo finale semplificato



Coeffice

	q <sub>2</sub>	q <sub>1</sub>	q <sub>0</sub>	Uscita
A*	0	0	0	0
B <sub>0</sub>	0	0	1	0
B <sub>1</sub>	0	1	1	0
C <sub>01</sub>	1	1	0	0
C <sub>10</sub>	0	1	0	0
C <sub>11</sub>	1	0	1	0
E	1	0	0	1

Architettura



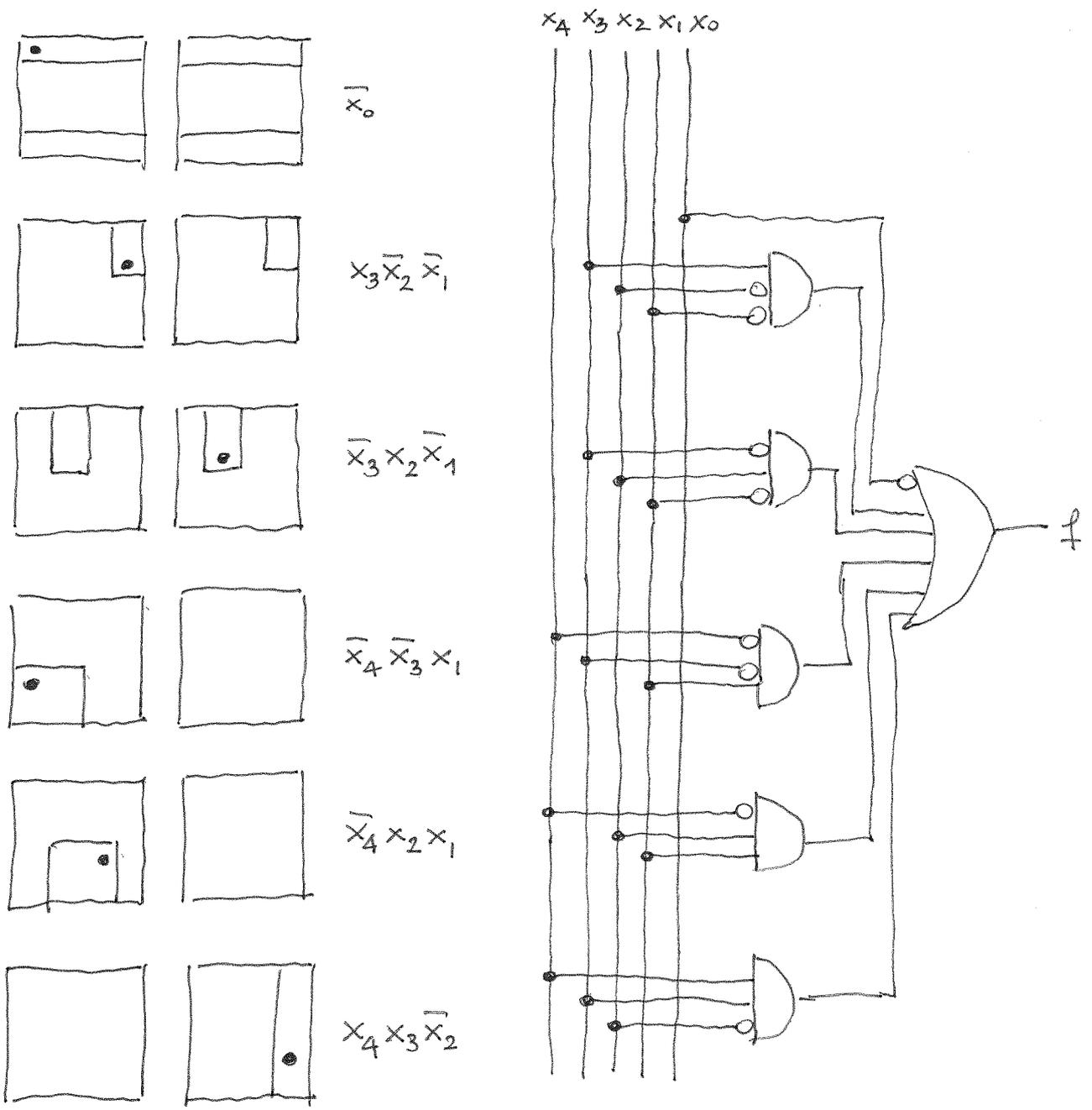
4

$x_4 = 0$

$x_3 x_2$	00	01	11	10
$x_1 x_0$ 00	1	4	12	8
01	0	5	13	9
11	3	7	15	11
10	2	6	14	10

$x_4 = 1$

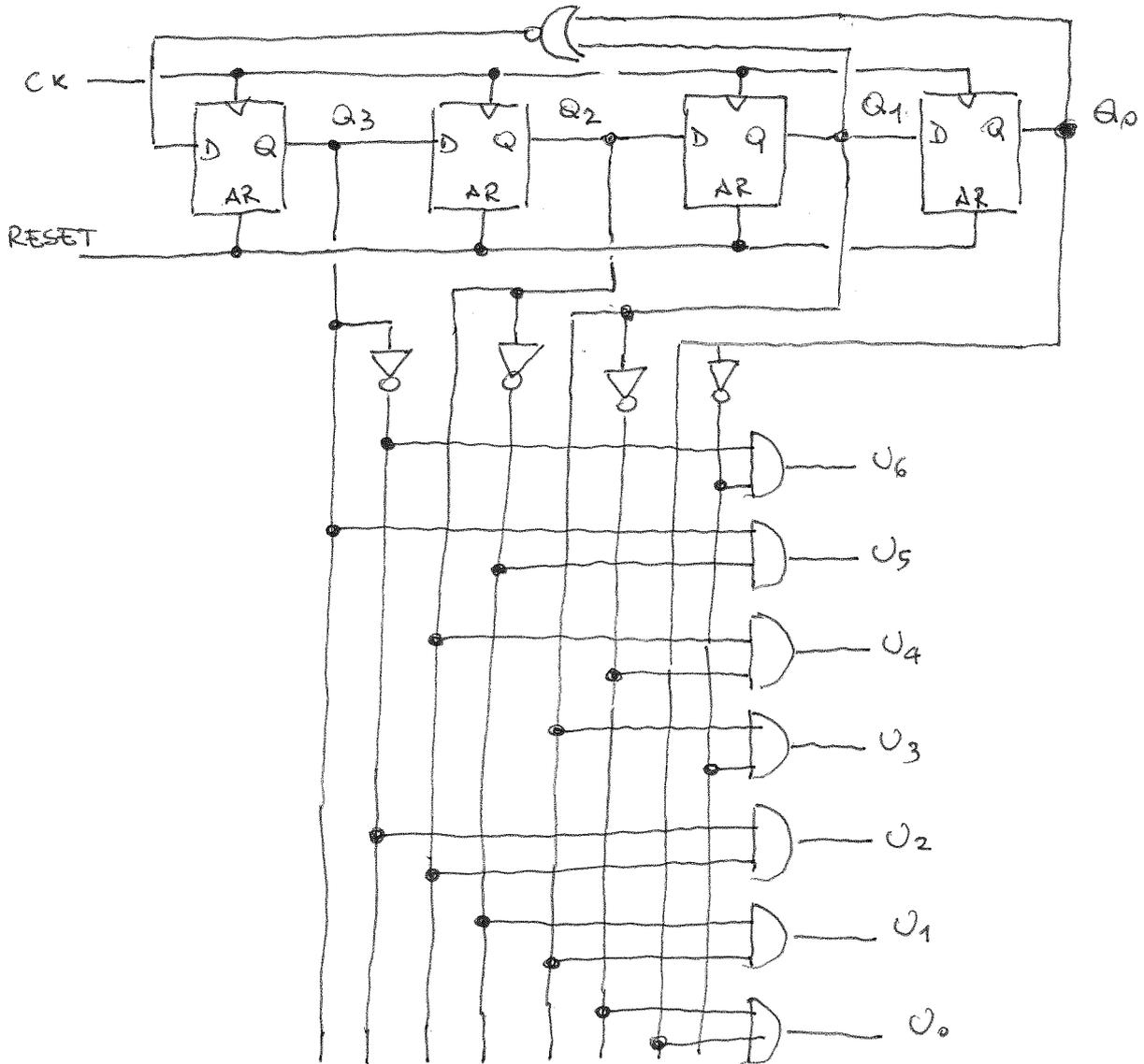
	00	01	11	10
	16	20	28	24
	17	21	29	25
	19	23	31	27
	18	22	30	26



Gli implicanti precedenti sono tutti ESSENZIALI. E' evidenziato un MINTERMINE (con il pallino) coperto unicamente dal singolo implicante principale indicato.

L'insieme degli essenziali COPRE TUTTI GLI 1 DELLA FUNZIONE, E' quindi la forma SP ottima.

5



Sequenze prodotte (dal reset)

$Q_3$	$Q_2$	$Q_1$	$Q_0$	$U_6$	$U_5$	$U_4$	$U_3$	$U_2$	$U_1$	$U_0$
0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0
1	1	0	0	0	0	1	0	0	0	0
1	1	1	0	0	0	0	1	0	0	0
0	1	1	1	0	0	0	0	1	0	0
0	0	1	1	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	0	1

$$U_6 = \overline{Q_3} \overline{Q_0}$$

$$U_5 = Q_3 \overline{Q_2}$$

$$U_4 = Q_2 \overline{Q_1}$$

$$U_3 = Q_1 \overline{Q_0}$$

$$U_2 = \overline{Q_3} Q_2$$

$$U_1 = \overline{Q_2} Q_1$$

$$U_0 = \overline{Q_1} Q_0$$

riperte

6) Nella notazione frazionale 1.7, il numero rappresentato ha LSB = 1/128.

a)

Quindi, detti A e B i due valori

$$A = \frac{17}{11} = \frac{197,818...}{128} \text{ e arrotondando } \frac{198}{128} \quad (\hat{A})$$

rappresentazione di A

1.1000110

$$B = \frac{178}{129} = \frac{176,620...}{128} \text{ e arrotondando } \frac{177}{128} \quad (\hat{B})$$

rappresentazione di B

1.0110001

Errori assoluti:  $|\hat{x} - x|$

$$\epsilon_A = 1,42 \cdot 10^{-3}$$

$$\epsilon_B = 2,97 \cdot 10^{-3}$$

(entrambi in eccesso)

Errori relativi:  $|(\hat{x} - x) / x|$

$$\epsilon_A\% = 0,0919\%$$

$$\epsilon_B\% = 0,215\%$$

b) L'istruzione FMUL moltiplica due operandi (1.7) e riporta il risultato a 16b nel formato (1.15) a partire dal (2.14) della normale moltiplicazione con una shift a SX. 18 bit più significativo, che esce, finisce nel CARRY.

Quindi

