

| | | |
|-----------------------|------|------------------------------|
| SCHEDA ASE1601 | | Data: 15 Gennaio 2016 |
| Cognome | Nome | |

ESERCIZIO N°1

8 punti

Nella memoria di un microcontrollore XMEGA della famiglia AVR, a partire dall'indirizzo il cui valore è contenuto nel puntatore Z, sono contenuti 3 valori senza segno da 1 byte che rappresentano la lunghezza dei 3 lati di un triangolo AB, BC e CA.

Realizzare una subroutine che determina se l'angolo ABC è acuto, retto oppure ottuso lasciando nel registro R16 rispettivamente i valori (C2) -1, 0, 1.

Nel caso in cui i 3 segmenti specificati non possano costituire un triangolo, in R16 va lasciato il valore 2.

ESERCIZIO N°2

5 punti

Disegnare il grafo di flusso e quindi sintetizzare secondo l'architettura di Moore una macchina sequenziale sincrona, con il minimo numero di flip-flop, con un ingresso e una uscita, in grado di riconoscere una qualsiasi tra le sequenze comunque interallacciate 101, 000 e 010. Ogni volta che una sequenza valida è riconosciuta, l'uscita è posta a 1 per un ciclo di clock.

ESERCIZIO N°3

4 punti

Progettare un contatore sincrono (up/down con abilitazione) modulo 13 facendo uso di T-FF.

ESERCIZIO N°4

6 punti

- a) Determinare la mappa di Karnaugh di una funzione logica $Y_{ok} = f(X_4, X_3, X_2, X_1, X_0)$ dove $X_3, X_2,$ e X_1, X_0 rappresentano due cifre consecutive in codifica GRAY a 2 bit; X_4 è un bit di parità con logica di Parità PARI. Y_{ok} vale 1 se si verificano queste due condizioni: la codifica GRAY è rispettata e la regola di parità è corretta. Y_{ok} vale 0 altrimenti.
- b) Realizzare con circuito a porte logiche AND-OR-NOT e 2 livelli di logica la funzione del punto a) scegliendo tra diverse possibili soluzioni quella che minimizza numero di porte logiche.
- c) Se porte logiche elementari (AND, OR, NOT) a K ingressi hanno $T_{pd} = 0,05 \text{ ns} + 0,15 K \text{ ns}$ quale è il T_{pd} massimo del circuito di cui al punto b)? Se ingressi e uscite del circuito combinatorio di cui al punto b) sono registrati con registri aventi $T_{co} = 0,4 \text{ ns}, T_{hold} = 0,2 \text{ ns}, T_{setup} = 0,4 \text{ ns}$ quale è la massima frequenza di lavoro possibile?

ESERCIZIO N°5

4 punti

Un circuito elettronico per telecomunicazioni che connette 8 unità che trasmettono e 32 unità che ricevono, è composto dalla cascata di un multiplexer 8 to 1 e di un demultiplexer 1 to 32:

- a) Realizzare il circuito usando decoder e porte three-state.
- b) Realizzare il circuito usando mux 2 to 1 e demux 1 to 2.

ESERCIZIO N°6

6 punti

Dati i numeri $A = -128,75$ $B = 1,125$ e i risultati delle operazioni $C = A + B$ e $D = A - B$

- a) Determinare la loro rappresentazione in virgola fissa e MS, C2, C1, Traslazione e il numero minimo di bit necessario per rappresentarli tutti correttamente
- b) Se si usa una ALU a 8 bit che opera in C2 si commettono errori di rappresentazione per A, B, C e D? Se si di che entità sono gli errori in valore assoluto e percentuale?
- c) Determinare la rappresentazione di A, B, C e D in virgola mobile formato standard IEEE 754 singola precisione. Si commettono errori di rappresentazione? Di che entità sono gli eventuali errori in valore assoluto e percentuale?

1

Nella memoria di un microcontrollore XMEGA della famiglia AVR, a partire dall'indirizzo il cui valore è contenuto nel puntatore Z, sono contenuti 3 valori senza segno da 1 byte che rappresentano la lunghezza dei 3 lati di un triangolo AB, BC e CA. Realizzare una subroutine che determina se l'angolo ABC è acuto, retto oppure ottuso lasciando nel registro R16 rispettivamente i valori (C2) -1, 0, 1. Nel caso in cui i 3 segmenti specificati non possano costituire un triangolo, in R16 va lasciato il valore 2.

```
angle_test:
  push R0 //salva i registri usati
  push R1
  push R2
  push R3
  push R17
  push R18
  push R20
  push R21
  push R22

  ld R16,Z //a=AB
  ldd R17,Z+1 //b=BC
  ldd R18,Z+2 //c=CA
  mov R20,R16 //replica a
  mov R21,R17 //replica b
  mov R22,R18 //replica c

  add R20,R17 //a+b
  brcs c_ok //sicuramente a+b>c
  cp R18,R20 //dovrebbe essere c<a+b
  brcc ko //non può essere un triangolo
c_ok:
  add R21,R18 //b+c
  brcs a_ok //sicuramente b+c>a
  cp R16,R21 //dovrebbe essere a<b+c
  brcc ko //non può essere un triangolo
a_ok:
  add R22,R16 //c+a
  brcs b_ok //sicuramente c+a>b
  cp R17,R22 //dovrebbe essere b<c+a
  brcc ko //non può essere un triangolo
b_ok:
```

```

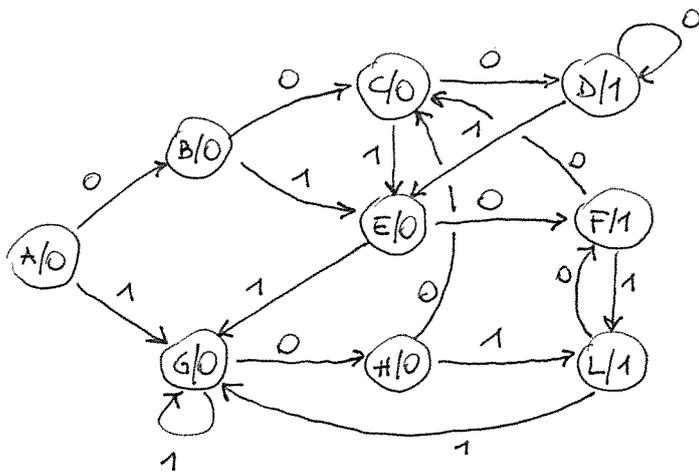
//ora analizza l'angolo
mul R16,R16 //a^2 in R1:R0
movw R3:R2,R1:R0
mul R17,R17 //b^2 in R1:R0
add R2,R0
adc R3,R1 //in R3:R2 ci va a^2+b^2, da confrontare con c^2
brcs ottuso //se c'è riporto l'angolo sarà ottuso
mul R18,R18 //c^2 in R1:R0
cp R2,R0
cpc R3,R1
breq retto //soddisfa teorema di Pitagora
brcc ottuso //a^2+b^2>c^2

acuto:
ldi R16,1
rjmp end_test_angle
retto:
ldi R16,0
rjmp end_test_angle
ottuso:
ldi R16,-1
rjmp end_test_angle
ko:
ldi R16,2

end_test_angle:
pop R22 //ripristina i registri usati, tranne R16 che contiene il risultato
pop R21
pop R20
pop R18
pop R17
pop R3
pop R2
pop R1
pop R0
ret

```

② Sequenze intersecciate 000, 010, 101



Per implementare questo grafo occorrono 4 D-F
Esaminiamo se ci sono stati equivalenti

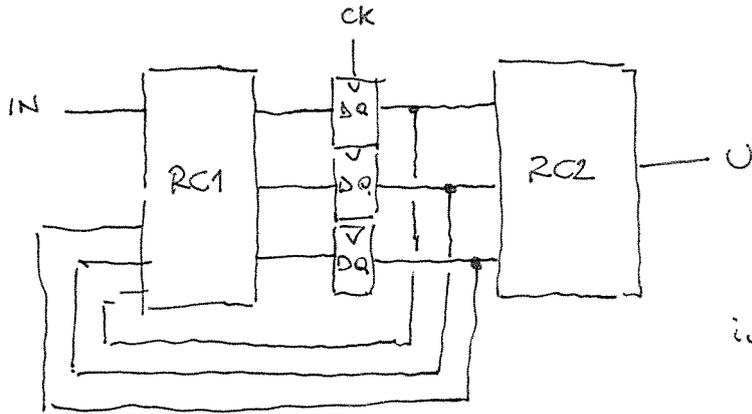
| | | | | | | | | | | |
|----------------|----------------|----------------|----------------|--------------|----------------|--------------|----------------|----------------|---|--|
| A ₀ | = | | | | | | | | | |
| B ₀ | C B E G | = | | | | | | | | |
| C ₀ | D B E X G | D C E X E | = | | | | | | | |
| D | X | X | X | = | | | | | | |
| E ₀ | F B G X G | F C G X E | F D G X E | X | = | | | | | |
| F | X | X | X | C D L X E | X | = | | | | |
| G ₀ | H B G X G | H C G X E | H D G X E | X | H F G X G | X | = | | | |
| H ₀ | C B L X G | C C L X E | C D L X E | X | C F L X G | X | C H L X G | = | | |
| L | X | X | X | F D G X E | X | F C G X L | X | X | = | |
| | A ₀ | B ₀ | C ₀ | D | E ₀ | F | G ₀ | H ₀ | L | |

Nel grafo NON ci sono stati equivalenti.

Potremmo ridurre a 3 il numero di D-F scegliendo arbitrariamente G₀ come stato iniziale.

In questo modo gli stati A e B non possono MAI essere raggiunti. Se prezzo da pagare è un possibile riconoscimento iniziale ERRATO nel caso in cui i primi 2 bit fossero 0,1

Si può quindi procedere alla sintesi con 3 D-FF con la seguente architettura



Codifica stati
(senza ottimizz.)

| | q_2 | q_1 | q_0 |
|-------------|-------|-------|-------|
| C | 0 | 1 | 1 |
| D | 1 | 1 | 1 |
| E | 0 | 0 | 1 |
| F | 1 | 0 | 1 |
| iniz. * G | 0 | 0 | 0 |
| H | 0 | 1 | 0 |
| L | 1 | 1 | 0 |
| non regg. M | 1 | 0 | 0 |

con la codifica scelta, $U = q_2$ e la RC_2 è un cortocircuito
Resta la sintesi di RC_1

| $q_1 q_0$ | IN, q_2 | | | |
|-----------|-----------|-----|-----|-----|
| | 00 | 01 | 11 | 10 |
| 00 | 010 | - | - | 000 |
| 01 | 101 | 011 | 110 | 000 |
| 11 | 111 | 111 | 001 | 001 |
| 10 | 011 | 101 | 000 | 110 |

| q_2 | 0 | 1 | 0 |
|-------|---|---|---|
| 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 |

$$q_2 = \bar{IN} \bar{q}_2 \bar{q}_0 + \bar{IN} q_2 q_1 + IN q_2 \bar{q}_1 + IN \bar{q}_2 q_1 \bar{q}_0$$

| q_1 | 1 | 1 | 0 |
|-------|---|---|---|
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |

$$q_1 = \bar{q}_2 q_1 \bar{q}_0 + \bar{IN} q_1 q_0 + \bar{IN} \bar{q}_2 \bar{q}_0 + q_2 \bar{q}_1$$

| q_0 | 0 | 1 | 0 |
|-------|---|---|---|
| 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 |

$$q_0 = \bar{IN} q_0 + \bar{IN} q_1 + q_1 q_0$$

Note: gli stati futuri di M sono H e L e non pongono problemi di lockout

④ a) Mappa Yok

| | | | | |
|-----------|-----------|----|----|----|
| | $x_3 x_2$ | | | |
| $x_1 x_0$ | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 0 | 0 |
| 01 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 |

$$x_4 = 0$$

| | | | | |
|-----------|-----------|----|----|----|
| | $x_3 x_2$ | | | |
| $x_1 x_0$ | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 0 | 1 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 0 | 1 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 |

$$x_4 = 1$$

Nota: l'insieme di 2 cifre che soddisfano la condizione di adiacenza in un codice Gray ha sempre un numero DISPARI di 1 (quindi per avere parità pari $x_4 = 1$)

- b) Sintesi: SP servono 8 AND a 5 ingressi
 1 OR a 8 ingressi
 PS servono 8 OR a 4 ingressi
 1 AND a 3 ingressi

(ottimo secondo il minimo numero di letterali;
 equivalenti dal punto di vista del NUMERO di porte)

Privilegio la PS, che ha meno letterali (33)
 Schema a pagina seguente

- c) il percorso critico prevede NOT, OR₄, AND₅

$$t_{pd NOT} = 0,20 \text{ ns}$$

$$t_{pd} = 2,25 \text{ ns}$$

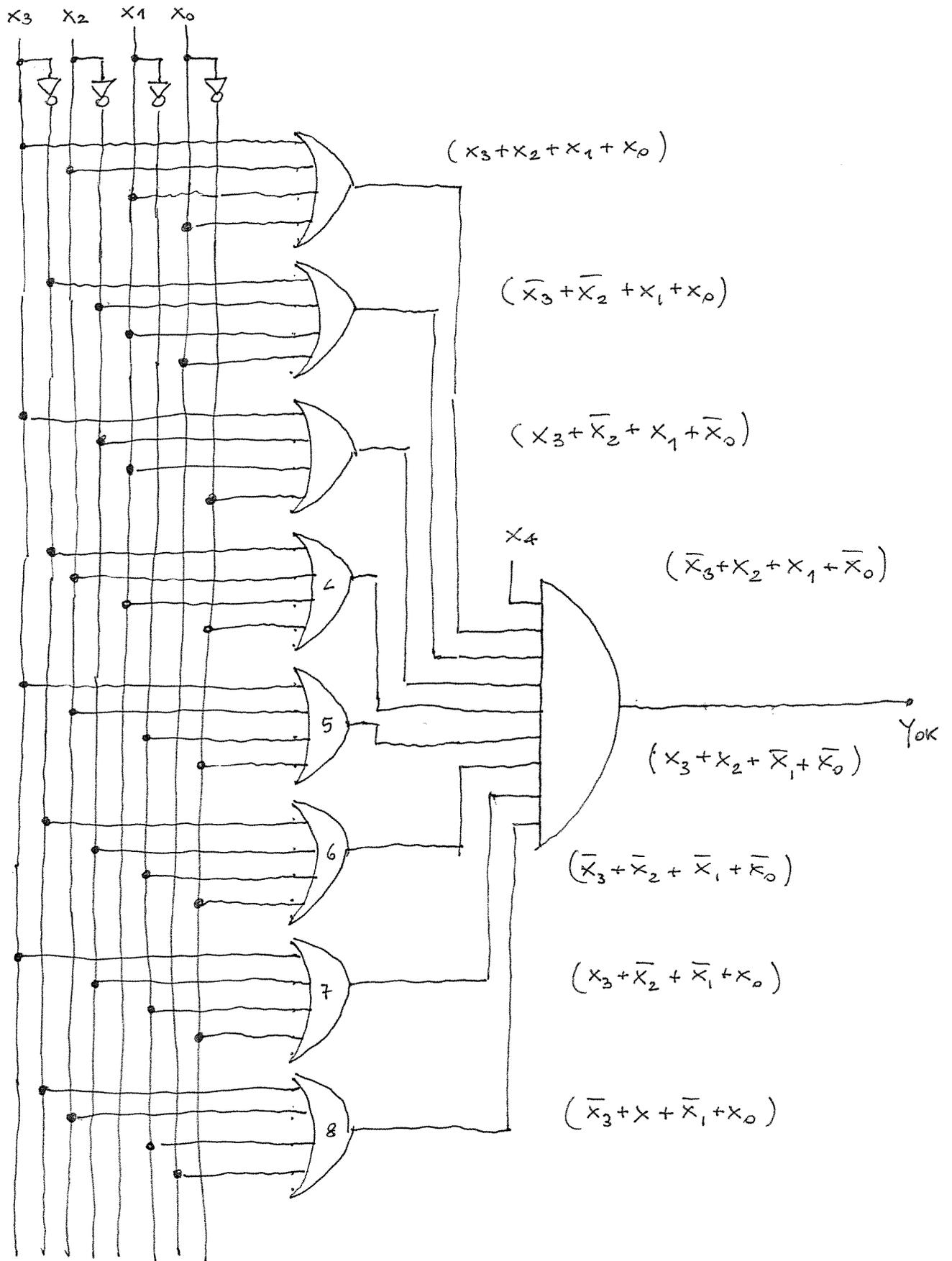
$$t_{pd OR4} = 0,65 \text{ ns}$$

$$t_{pd AND5} = 1,40 \text{ ns}$$

In architetture ripisizate si ha
 nel nostro caso

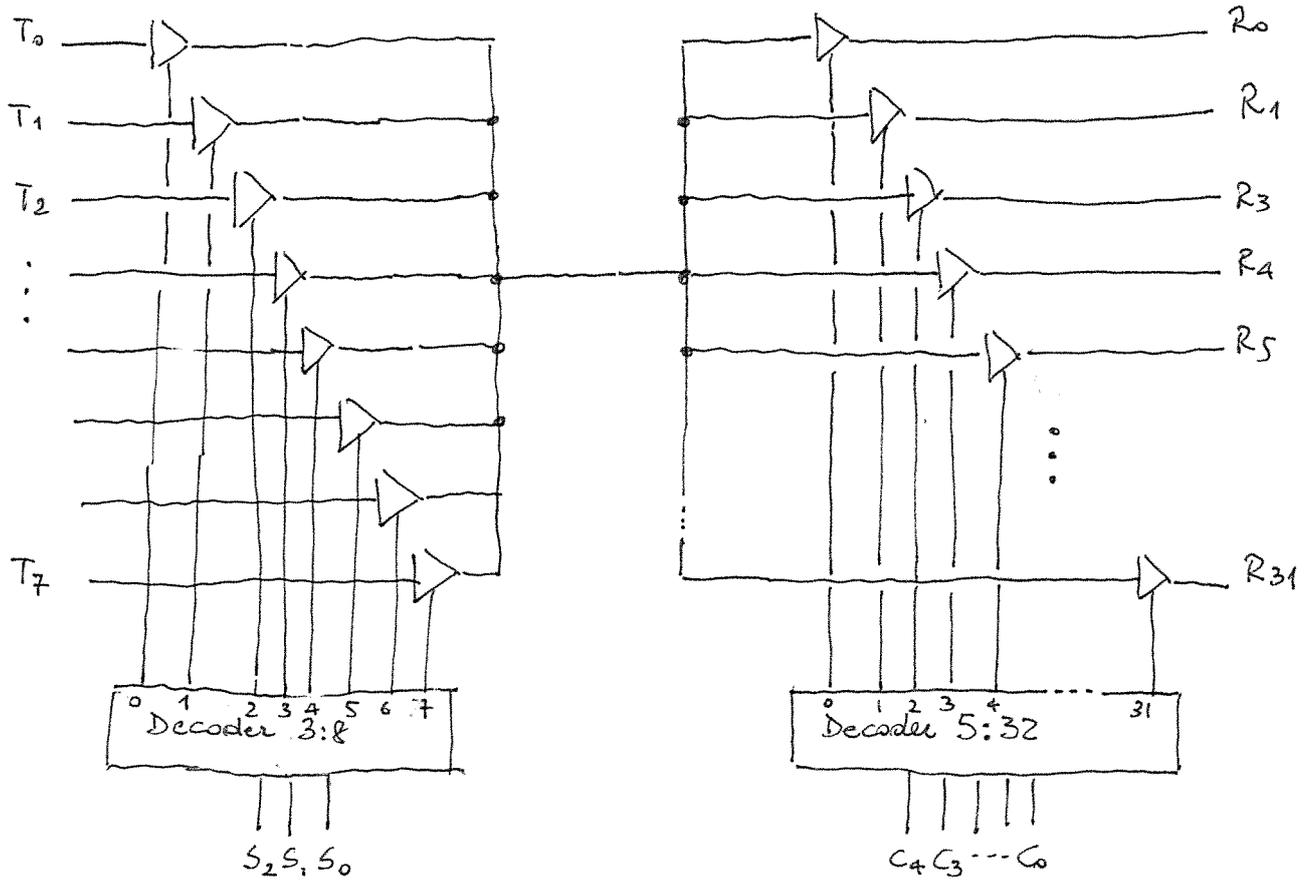
$$f_{max} = \frac{1}{t_{pd} + t_{su} + t_{co}}$$

$$f_{max} = 327,9 \text{ MHz}$$

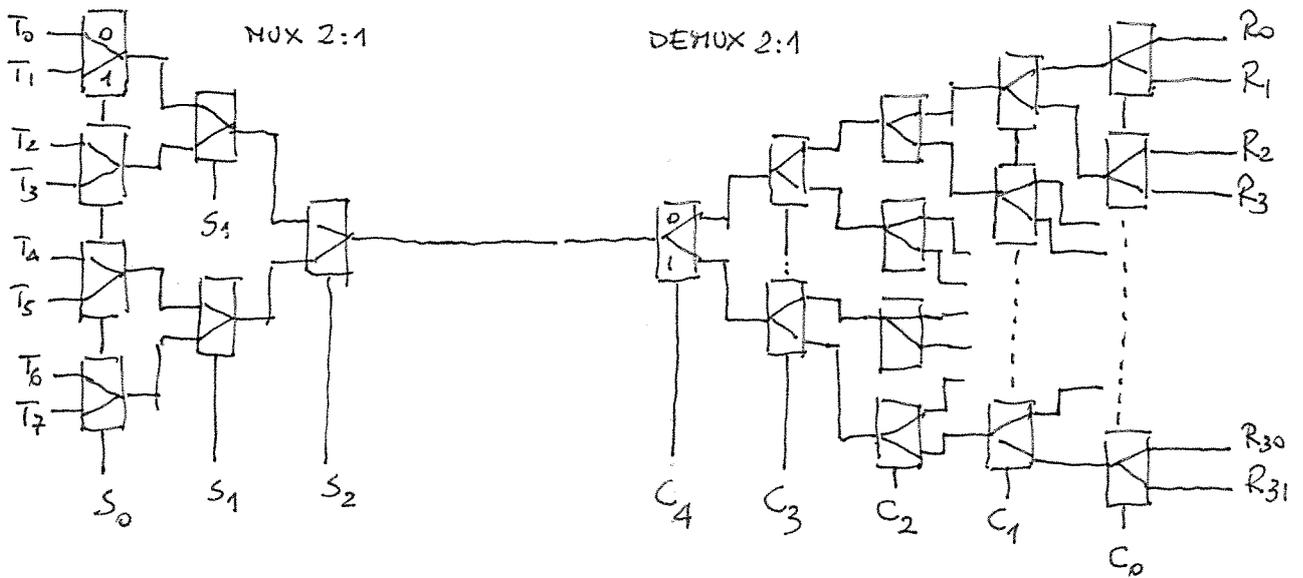


5

a) sezione di trasmissione - sezione di ricezione



b)



6

numeri da rappresentare

a) -128,75 ; 1,125 ; -127,625 ; -129,875

Le parti frazionarie possono essere rappresentate senza errore su 3 bit (6/8 ; 1/8 ; 5/8 ; 7/8)

Le parti intere, al segno, richiedono 9 bit in tutte le rappresent. indicate. In totale occorrono 12 bit.

-128,75 MS 1:10000000,110
 C1 10111111,001
 C2 10111111,010
 T 00111111,010

1,125 MS 0:00000001,001
 C1 00000000,001
 C2 00000000,001
 T 10000000,001

-127,625 MS 1:01111111,101
 C1 11000000,010
 C2 11000000,011
 T 01000000,011

-129,875 MS 1:10000001,111
 C1 10111110,000
 C2 10111110,001
 T 00111110,001

b) Usando 8 bit (eliminando i 4 bit meno significativi) si introduce errore. In C2 si ha. (senza troncamento)

| | | | | |
|----------|-----------------|-------|----------------------|------------------------|
| -128,75 | 1 0 1 1 1 1 1 1 | 0,000 | $\epsilon_A = 1,125$ | $\epsilon_r = 0,874\%$ |
| 1,125 | 0 0 0 0 0 0 0 0 | 0,000 | $\epsilon_A = 1,125$ | $\epsilon_r = 100\%$ |
| -127,625 | 1 1 0 0 0 0 0 0 | 0,000 | $\epsilon_A = 0,375$ | $\epsilon_r = 0,294\%$ |
| -129,875 | 1 0 1 1 1 1 1 1 | 0,000 | $\epsilon_A = 0,125$ | $\epsilon_r = 0,096\%$ |

c) La notazione IEEE 754, disponendo di un numero di cifre significative binarie maggiore di 12, rappresenta senza errore i valori esepuati:

$$x = (-1)^s 2^{e-127} \left\{ 1 + \sum_{i=1}^{23} b_{23-i} 2^{-i} \right\}$$

$$A = -128,75 = (-1)^1 2^7 (1 + 49152 \cdot 2^{-23})$$

$$s \quad e=134$$

$$[1; 10000110; 000.0000.1100.0000.0000.0000]$$

$$B = 1,125 = (-1)^0 2^0 (1 + 2^{20} \cdot 2^{-23})$$

$$s \quad e=127$$

$$[0; 01111111; 001.0000.0000.0000.0000.0000]$$

$$C = -127,625 = (-1)^1 2^6 (1 + 8339456 \cdot 2^{-23})$$

$$s \quad e=133$$

$$[1; 10000101; 111.1111.0100.0000.0000.0000]$$

$$D = -129,875 = (-1)^1 2^7 (1 + 122880 \cdot 2^{-23})$$

$$s \quad e=134$$

$$[1; 10000110; 000.0001.1110.0000.0000.0000]$$