

Elettronica Applicata  
*definizioni, formule ed esempi*

Pietro Barbiero

Quest'opera contiene informazioni tratte da wikipedia (<http://www.wikipedia.en>) e dalle dispense relative al corso di Elettronica Applicata e Misure tenuto dal professor Del Corso Dante del Dipartimento di Automatica e Informatica del Politecnico di Torino (IT).



Quest'opera è stata rilasciata con licenza Creative Commons Attribuzione - Non commerciale - Condividi allo stesso modo 4.0 Internazionale. Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-nc-sa/4.0/>.

# Indice

<b>I</b>	<b>Circuiti digitali</b>	<b>9</b>
<b>1</b>	<b>Flip-flop</b>	<b>11</b>
1.1	Tipologie di circuiti logici	11
1.1.1	Circuiti combinatori	11
1.1.2	Circuiti sequenziali	11
1.2	Circuiti di memoria	11
1.2.1	Circuito con latch	11
1.2.2	Circuito edge-triggred	11
1.3	Flip-flop (FF)	12
1.3.1	Tipologie	12
1.3.2	Latch FF	12
1.3.2.1	FF SR asincrono	12
1.3.2.2	FF SR sincrono	12
1.3.2.3	FF Latch D	13
1.3.3	Edge-triggred FF	14
1.3.3.1	FF MS D	14
1.3.3.2	FF DDR D	14
1.3.3.3	FF JK	15
1.3.4	Confronto risposte FF	16
1.3.5	Clear	16
1.4	Parametri elettrici dei FF	16
1.4.1	Ritardi	16
1.4.1.1	Tempo di setup	16
1.4.1.2	Tempo di hold	17
1.4.1.3	Tempo di propagazione	17
1.4.1.4	Tempo di salita (raise time)	17
1.4.1.5	Tempo di discesa (fall time)	17
1.4.1.6	Tempo di jitter del clock	17
1.4.1.7	Periodo minimo di clock	17
1.4.1.8	Frequenza massima di clock	17
1.4.2	Metastabilità	17
1.5	Esercizi	18
1.5.1	Risposta dei flip-flop	18
<b>2</b>	<b>Registri e contatori</b>	<b>21</b>
2.1	Trasmissione di segnali	21
2.1.1	Trasmissione seriale	21
2.1.2	Trasmissione parallela	21
2.2	Registro	21
2.2.1	Registro PIPO	21
2.2.2	Registro SISO	21
2.2.3	Registro SIPO	21

2.2.4	Registro PISO . . . . .	21
2.3	Contatore . . . . .	22
2.4	Divisore . . . . .	22
<b>3</b>	<b>Comparatori di soglia</b>	<b>23</b>
3.1	Comparatore di soglia . . . . .	23
3.1.1	Rumore . . . . .	23
3.1.2	Comparatore di soglia con isteresi (ritardo) . . . . .	24
3.1.3	Differenze con l'amplificatore . . . . .	24
3.1.4	Parametri dei comparatori . . . . .	24
3.2	<i>Esercizi</i> . . . . .	26
3.2.1	<i>Comparatori</i> . . . . .	26
<b>4</b>	<b>Generatori di segnali</b>	<b>27</b>
4.1	Parametri dei segnali periodici continui . . . . .	27
4.2	Generatore di segnali . . . . .	27
4.2.1	Parametri dei generatori a onda rettangolare . . . . .	27
4.2.2	Parametri dei generatori a onda quadra . . . . .	28
4.3	Circuito monostabile . . . . .	28
4.4	Generatore di onda quadra . . . . .	28
<b>5</b>	<b>Logiche programmabili</b>	<b>31</b>
5.1	Costi di progetto e produzione . . . . .	31
5.1.1	Costo per prodotto . . . . .	31
5.2	Stili di progetto . . . . .	31
5.3	Classificazione dei circuiti digitali . . . . .	32
5.4	Field Programmable Gate Array (FPGA) . . . . .	32
5.4.1	Celle logiche programmabili . . . . .	32
5.4.1.1	Programmable Array Logic (PAL) . . . . .	32
5.4.1.2	Programmable Logic Array (PLA) . . . . .	33
5.4.1.3	Programmable ROM (PROM) . . . . .	33
5.4.2	Memorie per la programmazione delle celle . . . . .	33
<b>II</b>	<b>Interconnessioni</b>	<b>35</b>
<b>6</b>	<b>Introduzione</b>	<b>37</b>
6.1	Entità delle interconnessioni . . . . .	37
6.1.1	Interconnessione ideale . . . . .	37
6.2	Modello RC passa basso o lineare . . . . .	37
6.2.1	Ritardi . . . . .	38
6.2.1.1	Tempo di trasmissione . . . . .	38
6.2.1.2	Skew . . . . .	38
6.3	Modello a parametri concentrati . . . . .	38
6.4	Modello a linea di trasmissione . . . . .	38
6.4.1	Esempi di linee di trasmissione . . . . .	39
6.4.2	Onda incidente . . . . .	39
6.4.3	Onda di riflessione . . . . .	39
6.4.3.1	Coefficiente di riflessione . . . . .	39
6.4.3.2	Tensione sulla linea . . . . .	40
6.5	Topologie di connessione . . . . .	40
6.5.1	Prestazioni . . . . .	40

<b>7 Cicli di trasferimento</b>	<b>43</b>
7.1 Cicli di trasferimento	43
7.1.1 Modello Sor-Dest	43
7.1.2 Ciclo completo	43
7.1.3 Classificazione dei cicli di trasferimento	43
7.1.4 Protocolli base	43
7.2 Protocolli per cicli di trasferimento	44
7.2.1 Protocollo sincrono	44
7.2.2 Protocollo asincrono	45
<b>8 Collegamenti paralleli</b>	<b>47</b>
8.1 Protocolli di un sistema a BUS	47
8.2 Allocazione	47
8.2.1 Tecniche di gestione del bus	47
8.2.2 Arbitraggio	47
8.3 Indirizzamento	48
8.3.1 Slave	48
8.3.2 Selezione dello Slave	48
8.4 Prestazioni	48
8.4.1 Parametri del BUS	48
8.4.2 Tecniche per il miglioramento delle prestazioni	49
8.4.2.1 Source Synchronous Protocol	49
8.4.2.2 Multiplexed BUS	49
8.4.2.3 Embedded Clock	49
8.4.2.4 Double Data Rate (DDR) Cycle	49
8.4.2.5 Burst	49
8.5 Limiti dei collegamenti paralleli	49
<b>9 Collegamenti seriali</b>	<b>51</b>
9.1 Collegamento seriale	51
9.1.1 CLK/Data embedding	51
9.1.2 Caratteristiche dei collegamenti seriali	51
9.2 Simbolo	51
9.2.1 Costellazione di segnali	52
9.3 Parametri dei collegamenti	52
9.3.1 Bit rate	52
9.3.2 Baud rate	52
9.3.3 Efficienza	52
9.3.4 ISI	52
9.3.5 Sincronismo	53
9.4 Diagramma a occhio	53
9.4.1 Parametri di un diagramma ad occhio	53
9.5 Collegamenti seriali	54
9.5.1 Collegamenti seriali sincroni e asincroni	54
9.5.2 Tecniche di sincronizzazione	54
9.5.3 Codifiche seriali	55
9.6 Modulazione dei segnali analogici	55
<b>10 Integrità di segnale</b>	<b>57</b>
10.1 Diafonia	57
10.1.1 Tipologie	57
10.1.2 Soluzioni	57
10.2 Rumore di commutazione	58

10.2.1 Soluzioni . . . . .	58
----------------------------	----

### III Conversioni A/D/A 59

<b>11 Introduzione</b>	<b>61</b>
11.1 Segnali analogici e digitali . . . . .	61
11.1.1 Comunicazione A/D/A . . . . .	61
11.2 Conversione A/D . . . . .	62
11.2.1 Campionamento . . . . .	62
11.2.1.1 Alias . . . . .	62
11.2.1.2 Aliasing . . . . .	62
11.2.1.3 Filtro anti aliasing . . . . .	62
11.2.1.4 Rumore di alising . . . . .	62
11.2.1.5 Teorema di Nyquist-Shannon . . . . .	62
11.2.1.6 Hold . . . . .	63
11.2.2 Quantizzazione . . . . .	63
11.2.2.1 Errore di quantizzazione . . . . .	63
11.2.2.2 Rapporto segnale/rumore (SNR) . . . . .	63
11.2.2.3 Distribuzione di probabilità . . . . .	64
11.2.2.4 Potenza dell'errore . . . . .	64
11.2.2.5 ENOB . . . . .	64
<b>12 Convertitori D/A</b>	<b>65</b>
12.1 Parametri . . . . .	65
12.1.1 Caratteristica di conversione D/A . . . . .	65
12.1.2 Parametri di un convertitore D/A . . . . .	65
12.2 Circuiti per convertitori D/A . . . . .	66
12.2.1 Struttura dei convertitori . . . . .	66
12.2.2 Tecniche base di progettazione . . . . .	66

# Elenco delle figure

1.1	Latch . . . . .	11
1.2	Edge-triggered . . . . .	12
1.3	FF SR asincrono . . . . .	13
1.4	FF SR sincrono . . . . .	13
1.5	FF Latch D . . . . .	14
1.6	FF MS D . . . . .	15
1.7	FF DDR D . . . . .	15
1.8	FF JK . . . . .	16
1.9	Diagrammi temporali . . . . .	18
2.1	Registri . . . . .	22
3.1	Comparatori di soglia . . . . .	23
3.2	Comparatore di soglia con isteresi . . . . .	24
4.1	Circuito monostabile . . . . .	28
4.2	Generatore di onda quadra . . . . .	29
6.1	Modello lineare . . . . .	37
6.2	Modello a linea di trasmissione . . . . .	40
6.3	Prestazioni . . . . .	41
7.1	Tempo di scrittura . . . . .	44
9.1	Rappresentazione dei simboli . . . . .	52
9.2	Symbol rate e efficienza . . . . .	53
9.3	Diagramma ad occhio . . . . .	53
9.4	Clock embedding . . . . .	54
9.5	Codifiche seriali . . . . .	55
9.6	Modulazioni ASK e PSK . . . . .	56
11.1	Conversione A/D . . . . .	62
11.2	Filtro anti aliasing . . . . .	62
11.3	Campionamento . . . . .	63
11.4	Quantizzazione . . . . .	64
12.1	Caratteristica di conversione D/A . . . . .	65





# Parte I

## Circuiti digitali



# Capitolo 1

## Flip-flop

### 1.1 Tipologie di circuiti logici

#### 1.1.1 Circuiti combinatori

Un circuito combinatorio è un circuito logico in cui il valore delle uscite nell'istante  $t_0$  è funzione solo degli ingressi applicati all'istante  $t_0$

$$O(t_0) = f(I_i(t_0)) \quad (1.1)$$

#### 1.1.2 Circuiti sequenziali

Un circuito sequenziale è un circuito logico in cui il valore delle uscite nell'istante  $t_0$  è funzione: degli ingressi applicati all'istante  $t_0$  e degli ingressi applicati prima di  $t_0$  (bisogna memorizzarli)

$$O(t_0) = f(I_i(t_0), I_i(t_{-1}), \dots) \quad (1.2)$$

### 1.2 Circuiti di memoria

#### 1.2.1 Circuito con latch

Un circuito di memoria con latch è un circuito sensibile ai livelli stabili: memorizza i bit durante i due stati stabili del segnale di ENABLE

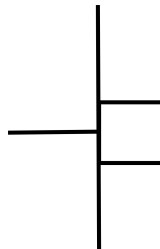


Figura 1.1: Latch

#### 1.2.2 Circuito edge-triggered

Un circuito di memoria edge-triggered è un circuito sensibile ai fronti di transizione: memorizza i bit durante le transizioni  $H \rightarrow L$  e/o  $L \rightarrow H$  del segnale di CLOCK

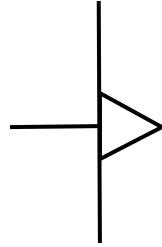


Figura 1.2: Edge-triggered

## 1.3 Flip-flop (FF)

Un flip-flop è un circuito sequenziale in grado di memorizzare uno stato logico

### 1.3.1 Tipologie

I FF possono essere suddivisi in due categorie:

- FF asincroni: possono cambiare stato in qualunque momento
- FF sincroni (con clock): possono cambiare stato solo in corrispondenza di un segnale di clock

### 1.3.2 Latch FF

#### 1.3.2.1 FF SR asincrono

Un FF SR (Set Reset) asincrono è un FF composto da due porte NOR; gli stati di un FF SR asincrono sono:

- stati di comando
  - SET ( $S = 1, R = 0$ )
  - RESET ( $S = 0, R = 1$ )
- stato di memoria ( $S = 0, R = 0$ )
- stato proibito ( $S = 1, R = 1$ )

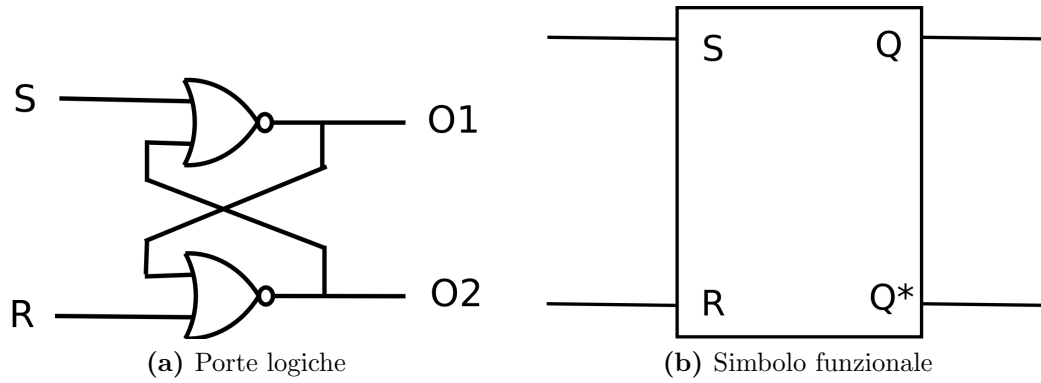
$S$	$R$	$Q$
1	0	1
0	1	0
0	0	$Q$
1	1	–

(1.3)

#### 1.3.2.2 FF SR sincrono

Un FF SR (Set Reset) sincrono è composto da due porte AND in ingresso ad un FF SR asincrono: ogni porta AND ha un ingresso di ENABLE e un ingresso o di SET o di RESET; gli stati di un circuito FF SR sincrono sono:

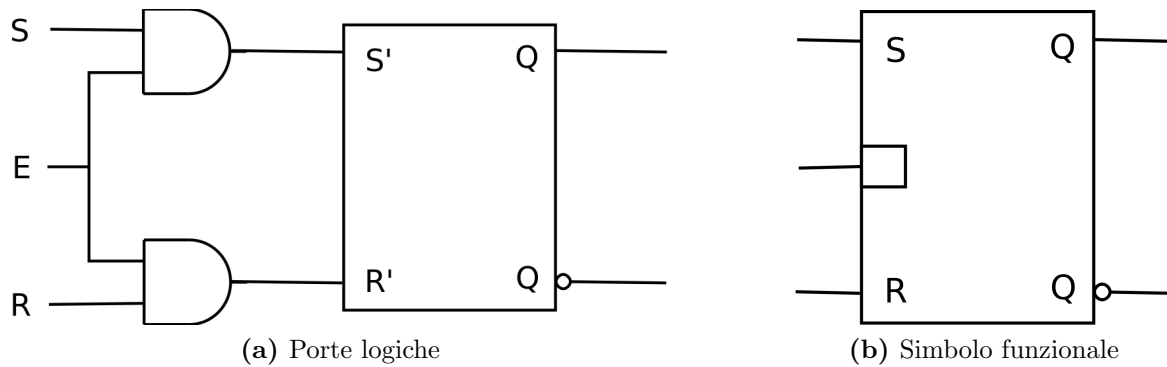
- stati di comando ( $E = 1$ )
  - SET ( $S = 1, R = 0$ )
  - RESET ( $S = 0, R = 1$ )

**Figura 1.3:** FF SR asincrono

- stato di memoria ( $E = 0$ )

$E$	$S$	$R$	$S'$	$R'$	$Q$
1	1	0	1	0	1
1	0	1	0	1	0
0	$x$	$x$	0	0	$Q$

(1.4)

**Figura 1.4:** FF SR sincrono

### 1.3.2.3 FF Latch D

Un FF Latch D è un FF composto da un unico ingresso D che va ai due ingressi S e R di un FF SR sincrono rispettivamente affermato e negato; gli stati di un circuito FF Latch D sono:

- stati di comando ( $E = 1$ )
  - SET ( $D = 1$ )
  - RESET ( $D = 0$ )
- stato di memoria ( $E = 0$ )

$D$	$E$	$S$	$R$	$S'$	$R'$	$Q$
1	1	1	0	1	0	1
0	1	0	1	0	1	0
$x$	0	$x$	$x$	0	0	$Q$

(1.5)

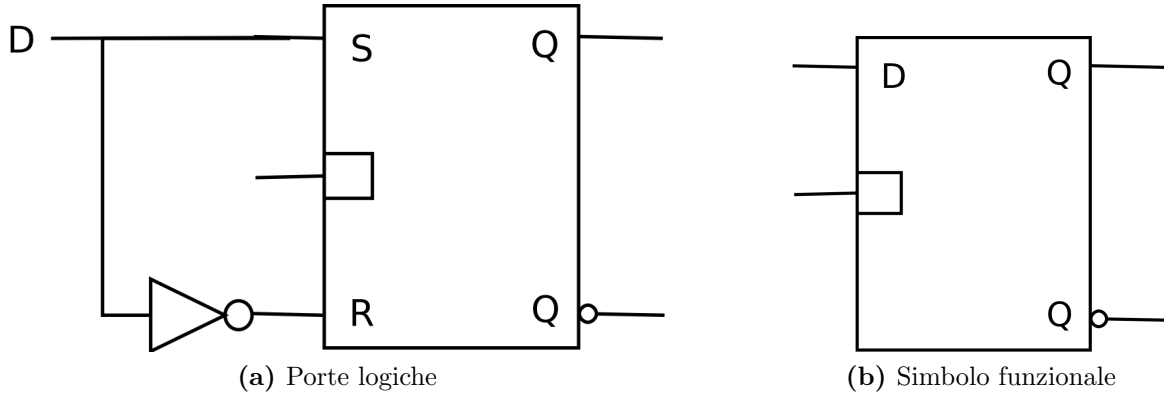


Figura 1.5: FF Latch D

### 1.3.3 Edge-triggered FF

#### 1.3.3.1 FF MS D

Un FF MS (Master-Slave) D è un FF edge-triggered composto da: un FF Latch D attivato dal segnale di clock negato (master) e un FF SR sincrono in serie (slave); gli stati di un circuito FF Master-Slave D sono:

- stati di comando
  - master ( $CLK = 0$ )
    - \* SET ( $D = 1$ )
    - \* RESET ( $D = 0$ )
  - slave ( $CLK = 1$ )
    - \* SET ( $Q_m = 1$ )
    - \* RESET ( $Q_m = 0$ )
- stato di memoria
  - master ( $CLK = 1$ )
  - slave ( $CLK = 0$ )

$D$	$CLK$	$Q_m$	$S$	$R$	$Q_s$
$x$	1	1	1	0	1
$x$	1	0	0	1	0
1	0	1	1	0	$Q_s$
0	0	0	0	1	$Q_s$

(1.6)

#### 1.3.3.2 FF DDR D

Un FF DDR (Double Data Rate) D è un FF edge-triggered composto da: due FF Latch D in parallelo con clock opposto; gli stati di un circuito FF DDR D sono:

- stati di comando
  - FF<sub>1</sub> ( $CLK = 1$ )
    - \* SET ( $D = 1$ )
    - \* RESET ( $D = 0$ )
  - FF<sub>2</sub> ( $CLK = 0$ )

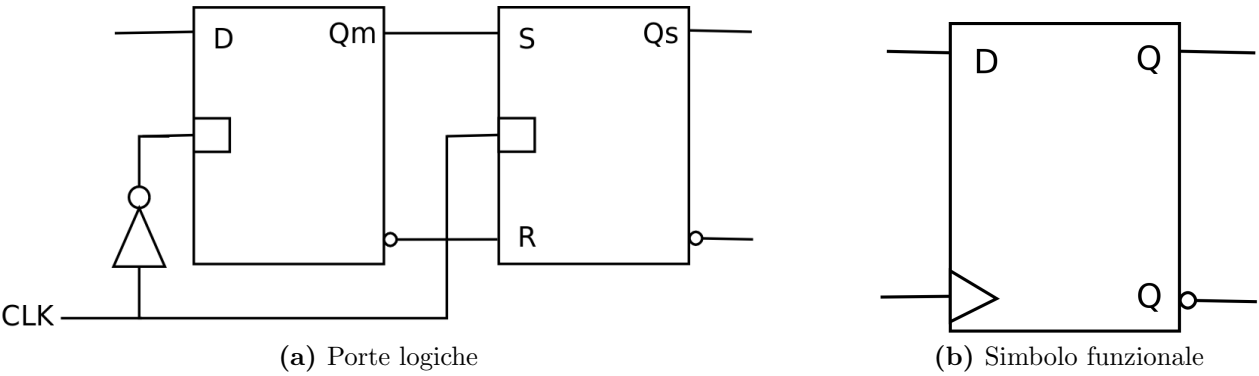


Figura 1.6: FF MS D

- \* SET ( $D = 1$ )
- \* RESET ( $D = 0$ )
- stati di memoria
  - $FF_1$  ( $CLK = 0$ )
  - $FF_2$  ( $CLK = 1$ )

$D$	$CLK$	$Q_1$	$Q_2$	$Q$
1	0	$Q_1$	1	$Q_1$
0	0	$Q_1$	0	$Q_1$
1	1	1	$Q_2$	$Q_2$
0	1	0	$Q_2$	$Q_2$

(1.7)

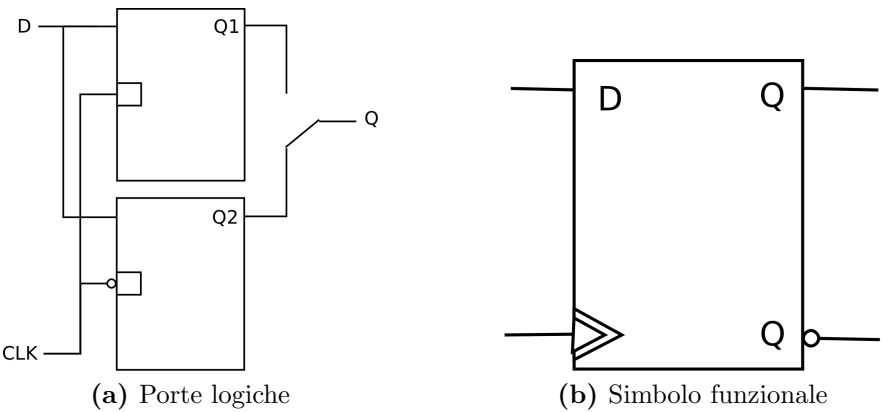


Figura 1.7: FF DDR D

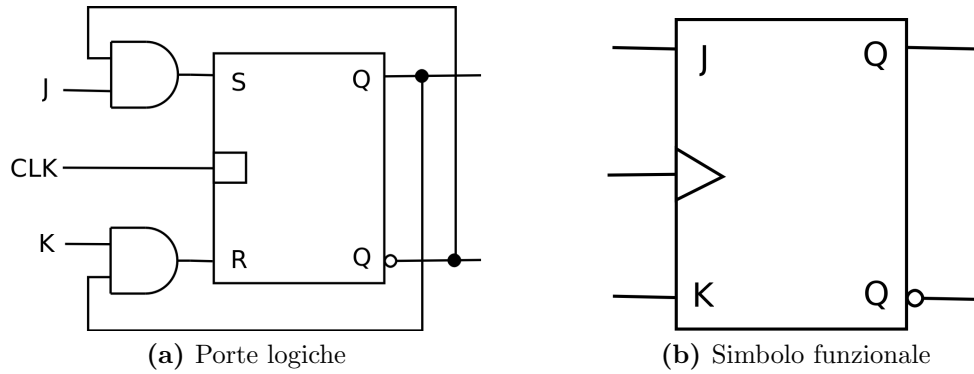
1.3.3.3 FF JK

Un FF JK è un FF edge-triggred composto da un FF SR con reazione incrociata; gli stati di un circuito FF JK sono:

- stati di comando
  - SET ( $J = 1, K = 0$ )
  - RESET ( $J = 0, K = 1$ )
- stato di memoria ( $J = 0, K = 0$ )

- stato di scambio ( $J = 1, K = 1$ )

$J$	$K$	$S$	$R$	$Q$	$Q^*$
1	0	$Q_{-1}^*$	0	1	0
0	1	0	$Q_{-1}$	0	1
0	0	0	0	$Q_{-1}$	$Q_{-1}^*$
1	1	$Q_{-1}^*$	$Q_{-1}$	$Q_{-1}^*$	$Q_{-1}$

(1.8)


**Figura 1.8:** FF JK

### 1.3.4 Confronto risposte FF

I FF memorizzano il bit di ingresso secondo tempistiche differenti:

- FF latch D: memorizza D quando l'ENABLE è alto
- FF master-slave D: memorizza D quando il CLOCK esegue la transizione  $L \rightarrow H$
- FF DDR D: memorizza D quando il CLOCK esegue le transizioni  $L \rightarrow H$  e  $H \rightarrow L$

### 1.3.5 Clear

Clear è un comando opzionale dei FF che permette di resettare a 0 l'uscita dei FF

## 1.4 Parametri elettrici dei FF

### 1.4.1 Ritardi

I ritardi nei circuiti con flip-flop sono di due tipologie:

- ritardo skew: ritardo tra una porta logica e l'altra
- ritardo jitter: ritardo relativo ad una singola porta logica dovuto al rumore

#### 1.4.1.1 Tempo di setup

Il tempo di setup  $t_s$  è l'intervallo di tempo minimo in cui il segnale di ingresso D di un FF deve rimanere stabile prima di un fronte di clock



**1.4.1.2 Tempo di hold**

Il tempo di hold  $t_h$  è l'intervallo di tempo minimo in cui il segnale di ingresso D di un FF deve rimanere stabile dopo un fronte di clock

**1.4.1.3 Tempo di propagazione**

Il tempo di propagazione  $t_p$  è l'intervallo di tempo che intercorre tra: l'istante in cui un segnale esce da una porta logica e l'istante in cui una porta logica a valle della prima riceve il segnale

**1.4.1.4 Tempo di salita (raise time)**

Il tempo di salita  $t_r$  è l'intervallo di tempo in cui una porta logica passa dallo stato  $L$  (10%) allo stato  $H$  90%

**1.4.1.5 Tempo di discesa (fall time)**

Il tempo di discesa  $t_f$  è l'intervallo di tempo in cui una porta logica passa dallo stato  $H$  (90%) allo stato  $L$  10%

**1.4.1.6 Tempo di jitter del clock**

Il tempo di jitter del clock  $t_j$  è un rumore temporale sul periodo di clock

**1.4.1.7 Periodo minimo di clock**

Il periodo minimo di clock  $T_{clk}$  è l'intervallo di tempo minimo sufficiente affinché le uscite e gli ingressi di un circuito siano stabili

$$T_{min} = t_s + t_h + t_p + t_j + \frac{t_f}{t_r} \quad (1.9)$$

**1.4.1.8 Frequenza massima di clock**

La frequenza massima di clock  $f_{clk}$  è l'inverso del periodo minimo di clock

$$f_{max} = \frac{1}{T_{min}} \quad (1.10)$$

**1.4.2 Metastabilità**

La metastabilità è una condizione in cui le uscite di un circuito non sono determinabili

## 1.5 Esercizi

### 1.5.1 Risposta dei flip-flop

**Esercizio 1.** Tracciare la risposta ai segnali  $D$  e  $CLOCK/ENABLE$  per i flip-flop  $D$  di tipo: latch, master-slave e DDR

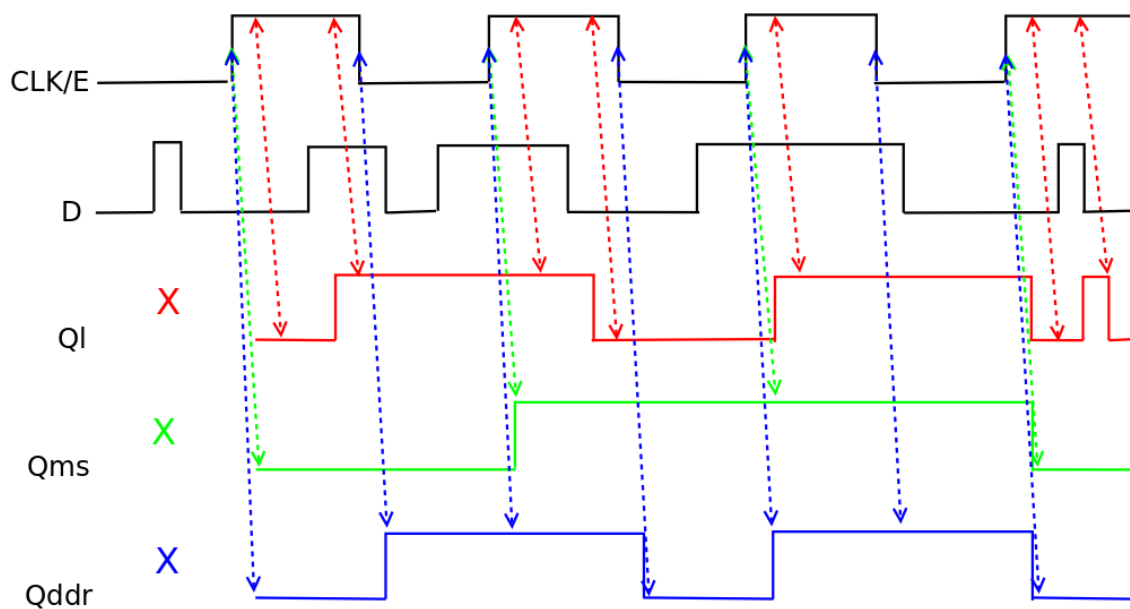
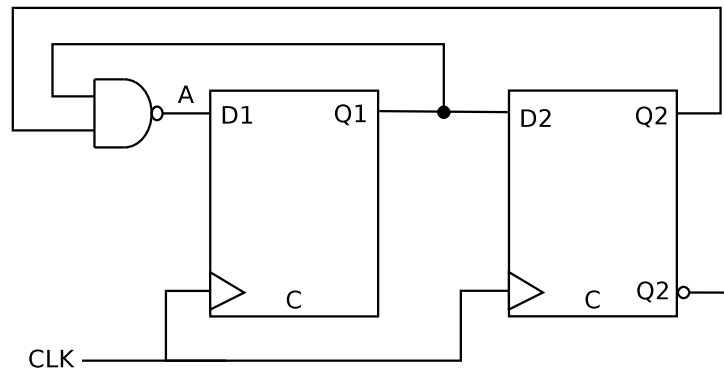
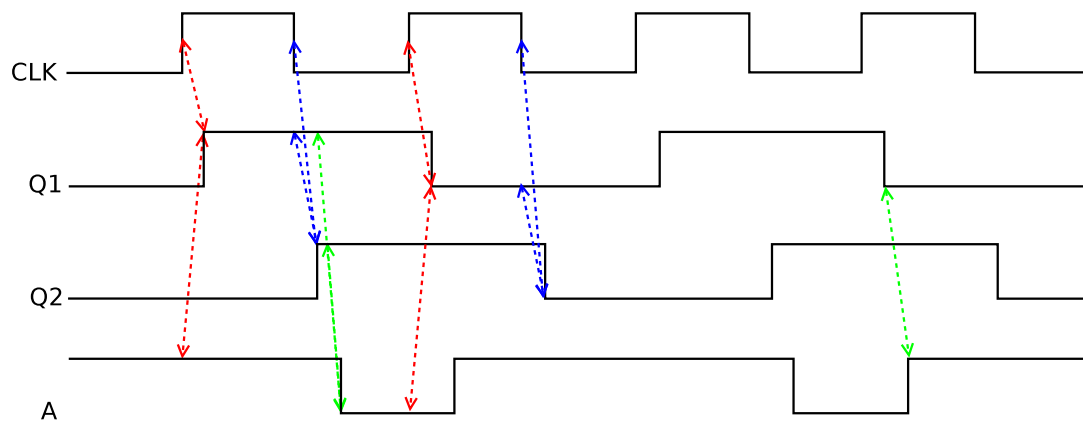


Figura 1.9: Diagrammi temporali

**Esercizio 2.** Analizzare il comportamento del circuito in figura supponendo che al tempo  $t_0$  le uscite siano resettate a 0 dal comando *CLEAR*; calcolare la frequenza massima del clock del primo FF dati:  $t_{CKQ} = 5 \text{ ns}$ ,  $t_{LH} = 3 \text{ ns}$  (AND),  $t_{HL} = 4 \text{ ns}$  (AND),  $t_s = 3 \text{ ns}$



1) analisi



2) frequenza massima del clock del primo FF

$$T_{min} = t_{CKQ} + t_{HL} + t_s = 12 \text{ ns} \implies f = \frac{1}{T_{min}} = 83.3 \text{ MHz} \quad (1.11)$$



# Capitolo 2

## Registri e contatori

### 2.1 Trasmissione di segnali

#### 2.1.1 Trasmissione seriale

La trasmissione seriale è una tecnica di trasmissione di segnali in cui i bit del segnale vengono inviati da un mittente uno di seguito all'altro e giungono al ricevente in sequenza nello stesso ordine di partenza; si possono trasmettere  $n$  bit con  $n$  cicli di clock; il segnale è distribuito nel tempo

#### 2.1.2 Trasmissione parallela

La trasmissione parallela è una tecnica di trasmissione di segnali in cui il mittente e il ricevente effettuano  $n$  trasmissioni seriali di  $n$  segnali su  $n$  canali; si trasmettono  $n$  bit (uno per segnale) ad ogni colpo di clock; il segnale è distribuito nello spazio

### 2.2 Registro

Un registro è un circuito elettronico in grado di memorizzare una serie di bit

#### 2.2.1 Registro PIPO

Un registro PIPO (Parallel Input Parallel Output) è un registro composto da  $n$  FF aventi ingressi e uscite in parallelo (trasmissione parallela)

#### 2.2.2 Registro SISO

Un registro SISO (Serial Input Serial Output) o shift-register è un registro composto da  $n$  FF aventi ingressi e uscite in serie (trasmissione seriale)

#### 2.2.3 Registro SIPO

Un registro SIPO (Serial Input Parallel Output) è un registro composto da  $n$  FF aventi ingressi in serie e  $n$  uscite in parallelo (conversione seriale  $\rightarrow$  parallelo)

#### 2.2.4 Registro PISO

Un registro PISO (Parallel Input Serial Output) è un registro composto da  $n$  FF aventi ingressi in parallelo e uscita in serie (conversione parallelo  $\rightarrow$  seriale)

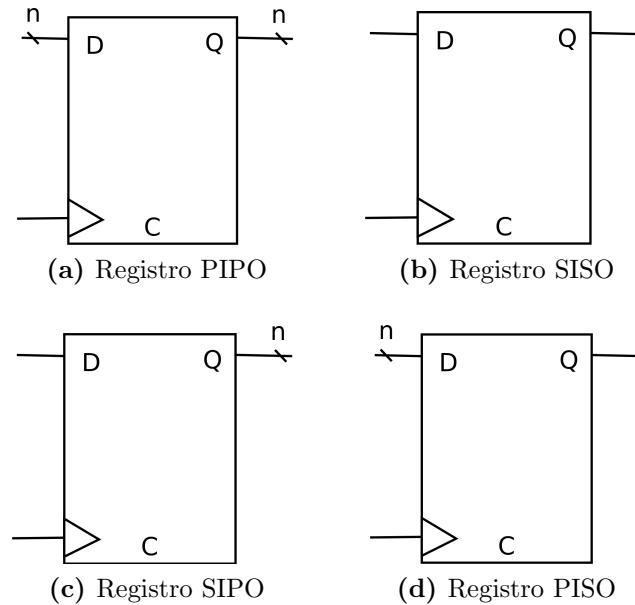


Figura 2.1: Registri

## 2.3 Contatore

Un contatore è un circuito elettronico in grado di contare il numero di volte che si verifica un determinato evento; esistono due tipologie di contatori:

- contatori UP: contano in verso crescente
- contatori DOWN: contano in verso decrescente

## 2.4 Divisore

Un divisore è un circuito elettronico che riceve in ingresso un segnale di frequenza  $f_I$  e restituisce in uscita un segnale di frequenza  $F_I/n$ ; un divisore può essere implementato attraverso  $n$  FF retroattivi in serie in cui l'uscita di ogni FF pilota il clock di quello successivo

# Capitolo 3

## Comparatori di soglia

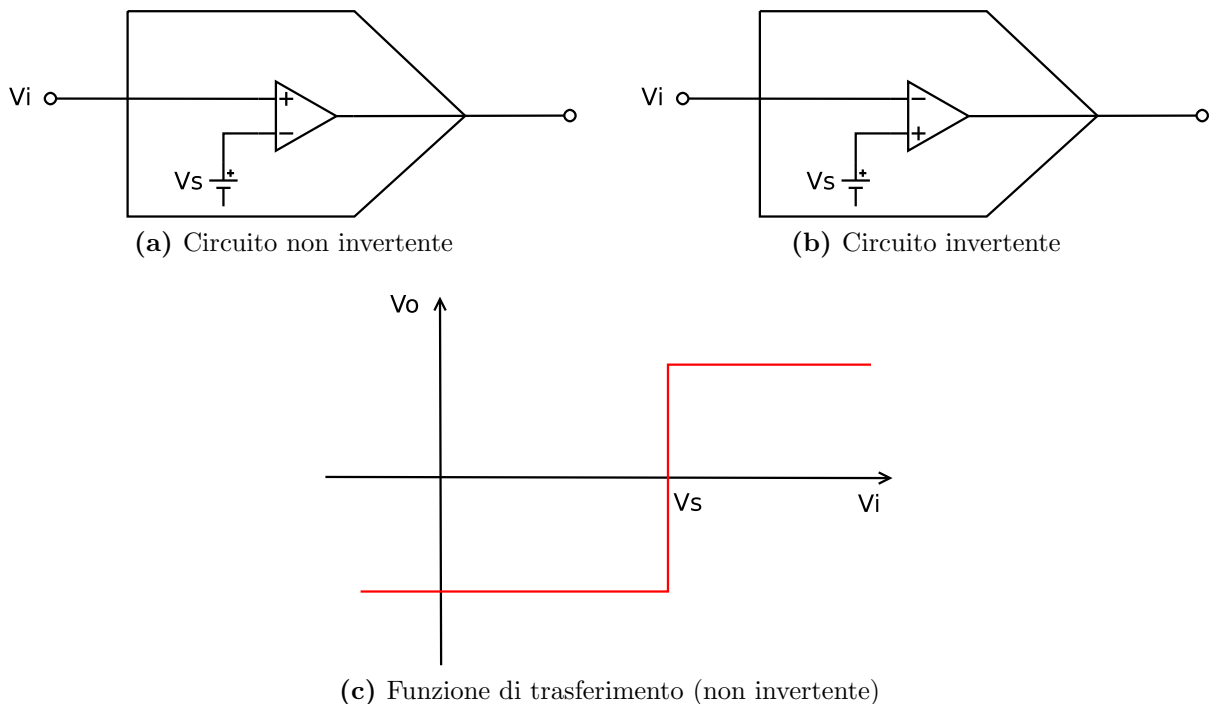
### 3.1 Comparatore di soglia

Un comparatore di soglia è un circuito elettronico che:

- riceve in ingresso un segnale analogico  $I$
- confronta il segnale in ingresso con una soglia  $S$
- restituisce in uscita un segnale binario  $O$

$$- I > S \implies O = H$$

$$- I < S \implies O = L$$



**Figura 3.1:** Comparatori di soglia

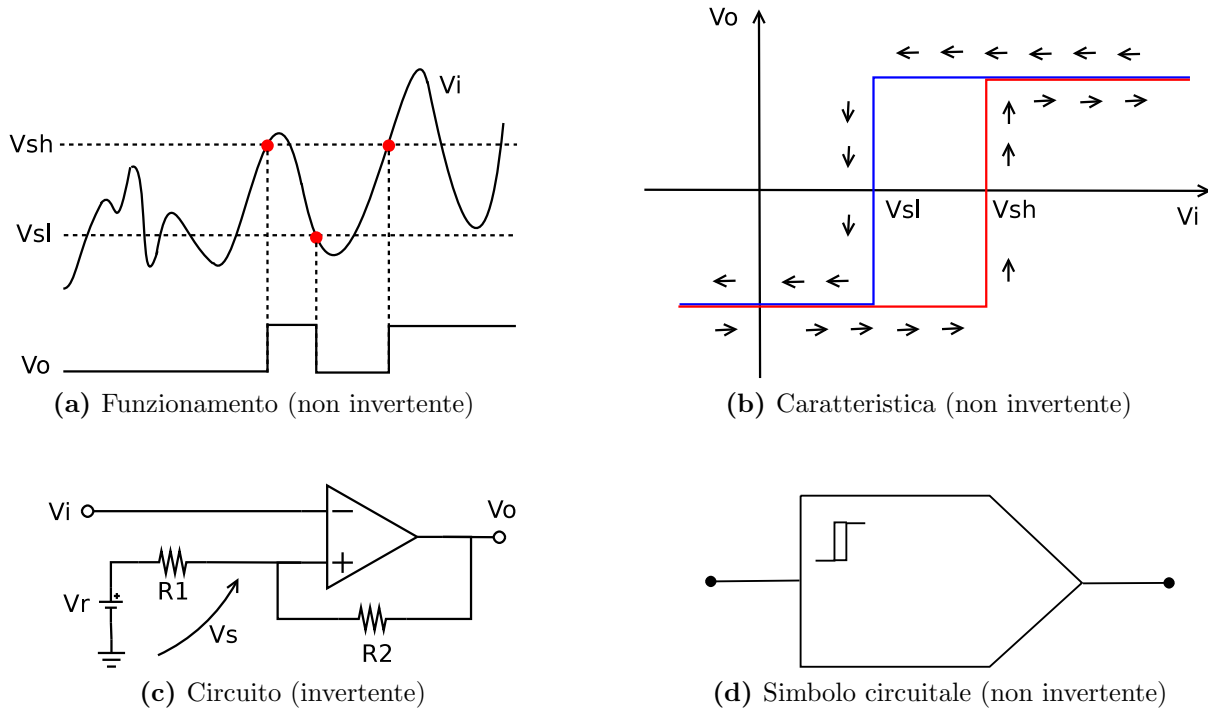
#### 3.1.1 Rumore

Il rumore presente nei segnali reali può causare involontariamente l'attraversamento multiplo della soglia

### 3.1.2 Comparatore di soglia con isteresi (ritardo)

Un comparatore di soglia con isteresi è un comparatore avente due soglie  $S_H$  e  $S_L$  in cui:

- l'uscita  $O$  commuta in  $H$  soltanto se: il segnale è crescente e attraversa  $V_{S_H}$
- l'uscita  $O$  commuta in  $L$  soltanto se: il segnale è decrescente e attraversa  $V_{S_L}$



**Figura 3.2:** Comparatore di soglia con isteresi

### 3.1.3 Differenze con l'amplificatore

Il comparatore ha le seguenti differenze con l'amplificatore:

- reazione sul morsetto positivo dell'OA (negativa per A)
- tensione differenziale di ingresso qualunque ( $\sim 0$  per A)
- commuta quando  $V_d = 0$
- uscita satura ai livelli di alimentazione (mai satura per A)

### 3.1.4 Parametri dei comparatori

I parametri dei comparatori di soglia sono:

- statici
  - tensioni differenziali e di modo comune massime (non danneggiamento)
  - correnti di polarizzazione (bias)
  - tensioni e correnti di offset

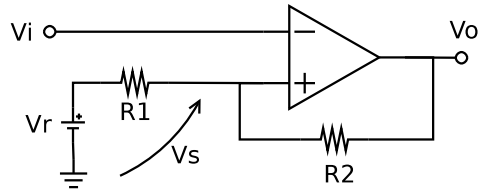


- uscita
  - tensioni di stato (livelli logici  $H - L$ )
  - correnti erogabili
- dinamici
  - tempi di salita  $t_r$  e discesa  $t_f$  (dal 10% al 90% della variazione)
  - ritardo di propagazione  $t_{p_{HL}}$  e  $t_{p_{LH}}$  (dalla ricezione del segnale al 50% della variazione)

## 3.2 Esercizi

### 3.2.1 Comparatori

**Esercizio 3.** Dato il comparatore in figura avente i seguenti parametri:  $R_1 = 22k\Omega$ ,  $R_2 = 120k\Omega$ ,  $V_R = 3V$ ,  $alim = \pm 15V$ ; determinare i valori delle soglie  $V_{s1}$  e  $V_{s2}$



1)  $V_O$

$$V_O \sim alim - 1 = \pm 14V$$

2)  $V_S$

$$V_{s1} = V_{OH} \frac{R_1}{R_1 + R_2} + V_R \frac{R_2}{R_1 + R_2} = 4.62V$$

$$V_{s2} = V_{OL} \frac{R_1}{R_1 + R_2} + V_R \frac{R_2}{R_1 + R_2} = 0.42V$$

# Capitolo 4

## Generatori di segnali

### 4.1 Parametri dei segnali periodici continui

I parametri dei segnali periodici continui sono:

- parametri del I ordine
  - forma d'onda (sinusoide, onda quadra, onda triangolare)
  - periodo  $T$ : intervallo di tempo in cui il segnale compie un'oscillazione completa e torna allo stato iniziale
  - frequenza  $f$ : numero di oscillazioni nell'unità di tempo

$$f = \frac{1}{T} \quad (4.1)$$

- ampiezza
  - \* ampiezza di picco  $A_p$ : massima variazione della grandezza dal valor medio
  - \* ampiezza picco-picco  $A_{pp}$ : massima escursione tra il punto di oscillazione più basso e il punto di oscillazione più alto
- duty cycle (ciclo di lavoro): rapporto tra il tempo  $\tau$  in cui il segnale è in uno stato attivo e il tempo  $T$  in cui viene effettuata l'osservazione

$$d = \frac{\tau}{T} \quad (4.2)$$

- parametri del II ordine
  - livello DC
  - distorsione

### 4.2 Generatore di segnali

Un generatore di segnali è un circuito elettronico in grado di generare un segnale elettrico predeterminato

#### 4.2.1 Parametri dei generatori a onda rettangolare

I parametri dei generatori di segnali a onda rettangolare (o generatori di impulsi) sono:

- livelli di tensione massimi di uscita  $V_H - V_L$
- periodo del segnale  $T = t_H + t_L$

- duty-cycle  $DC = t_H/T \neq 0.5$
- tempo di salita  $t_r$  e di discesa  $t_f$

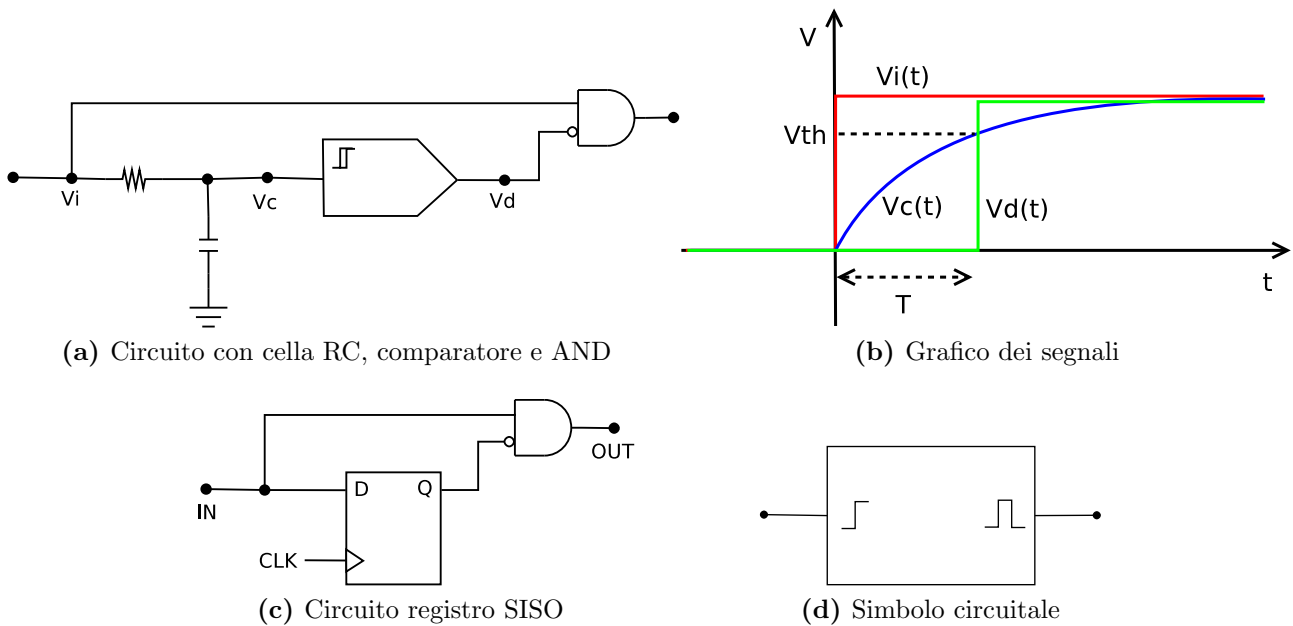
### 4.2.2 Parametri dei generatori a onda quadra

I parametri dei generatori di segnali a onda quadra sono:

- livelli di tensione massimi di uscita  $V_H - V_L$
- periodo del segnale  $T = t_H + t_L$
- duty-cycle  $DC = t_H/T = 0.5$
- tempo di salita  $t_r$  e di discesa  $t_f$

## 4.3 Circuito monostabile

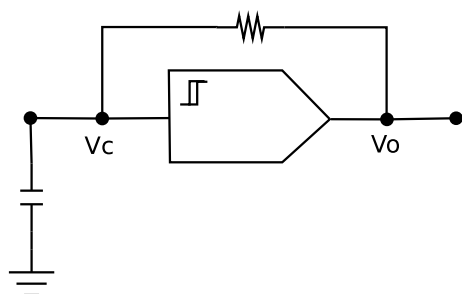
Un circuito monostabile è un generatore di segnali che genera un impulso di larghezza  $W$  in corrispondenza di una transizione all'ingresso



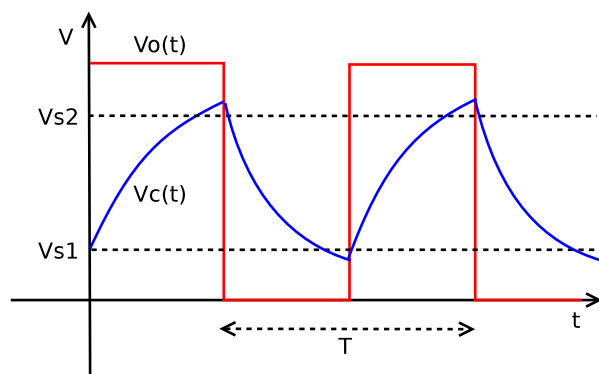
**Figura 4.1:** Circuito monostabile

## 4.4 Generatore di onda quadra

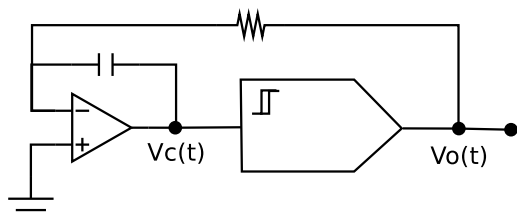
Un generatore di onda quadra è un generatore di segnali che genera un'onda quadra



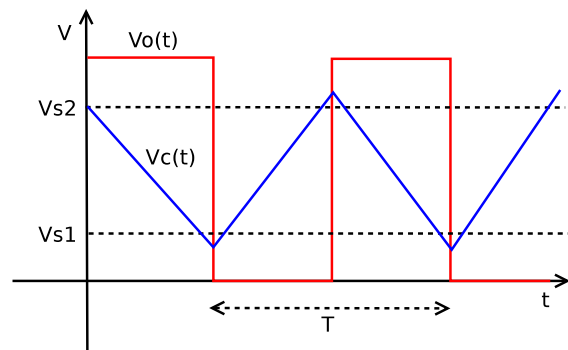
(a) Circuito con cella RC e comparatore



(b) Grafico dei segnali



(c) Circuito con integratore e comparatore



(d) Grafico dei segnali

**Figura 4.2:** Generatore di onda quadra



# Capitolo 5

## Logiche programmabili

### 5.1 Costi di progetto e produzione

I costi di progetto e produzione di un prodotto possono essere classificati in:

- Non-Recurring Engineering cost ( $NRE$ ): costo pagato una tantum: costo di progetto + costo degli impianti di produzione
- costo unitario di produzione ( $Cu$ ): costo per produrre un singolo oggetto: costo materiale + costo manodopera

#### 5.1.1 Costo per prodotto

Il costo per prodotto  $Cp$  è dato dalla somma tra: il costo unitario di produzione e il rapporto tra l' $NRE$  cost e il numero  $N$  di oggetti prodotti

$$Cp = \frac{NRE}{N} + Cu \quad (5.1)$$

### 5.2 Stili di progetto

Gli stili di progetto per i circuiti elettronici possono essere classificati in:

- commodity
  - circuiti Components Off-The Shelf (porte, registri memorie, microprocessori, periferiche,...)
  - $NRE$  molto basso
  - funzione completamente variabile
- processori
  - hardware generico con funzioni definite dal SO
  - $NRE$  basso
  - funzione variabile tramite programmazione software
- circuiti logici programmabili
  - circuiti prefabbricati
  - $NRE$  medio

- funzione variabile dall'utente
- semicustom
  - circuiti parzialmente prefabbricati
  - *NRE* medio-alto
  - funzione variabile dalle aziende
- custom
  - circuiti specializzati (transceiver, processori video,...)
  - *NRE* molto alto
  - funzione fissa

## 5.3 Classificazione dei circuiti digitali

I circuiti digitali possono essere classificati in:

- logiche standard (porte logiche, microprocessori)
  - ottimizzazione bassa
  - flessibilità massima
- logiche programmabili (PLD, FPGA, CPLD)
  - ottimizzazione media
  - flessibilità media
- Application Specific Integrated Circuit (circuiti per applicazioni specifiche)
  - ottimizzazione massima
  - flessibilità minima

## 5.4 Field Programmable Gate Array (FPGA)

Una FPGA è una logica programmabile complessa composta da:

- celle logiche programmabili
- interconnessioni programmabili
- celle di I/O programmabili

### 5.4.1 Celle logiche programmabili

#### 5.4.1.1 Programmable Array Logic (PAL)

Un circuito PAL è costituito da una matrice di AND programmabile e da una matrice di OR non programmabile



#### 5.4.1.2 Programmable Logic Array (PLA)

Un circuito PLA è costituito da una matrice di AND e da una matrice di OR programmabili

#### 5.4.1.3 Programmable ROM (PROM)

Un circuito PROM è costituito da una matrice di AND cablata come decoder (per la decodifica di indirizzi; contiene i collegamenti per tutti gli indirizzi possibili) non programmabile e da una matrice di OR programmabile

### 5.4.2 Memorie per la programmazione delle celle

Le informazioni di programmazione delle celle possono essere memorizzate in:

- memoria volatile (RAM, registri)
- memoria non volatile riprogrammabile (EPROM, EEPROM, FLASH)
- memoria non volatile non riprogrammabile (PROM)



# Parte II

## Interconnessioni



# Capitolo 6

## Introduzione

### 6.1 Entità delle interconnessioni

In una interconnessione tra due circuiti intervengono le seguenti entità:

- driver: è il circuito che trasmette il segnale attraverso l'interconnessione
- receiver: è il circuito che riceve il segnale trasmesso attraverso l'interconnessione
- interconnessione: è il mezzo trasmissivo attraverso il quale vengono trasferite informazioni sotto forma di segnali elettrici

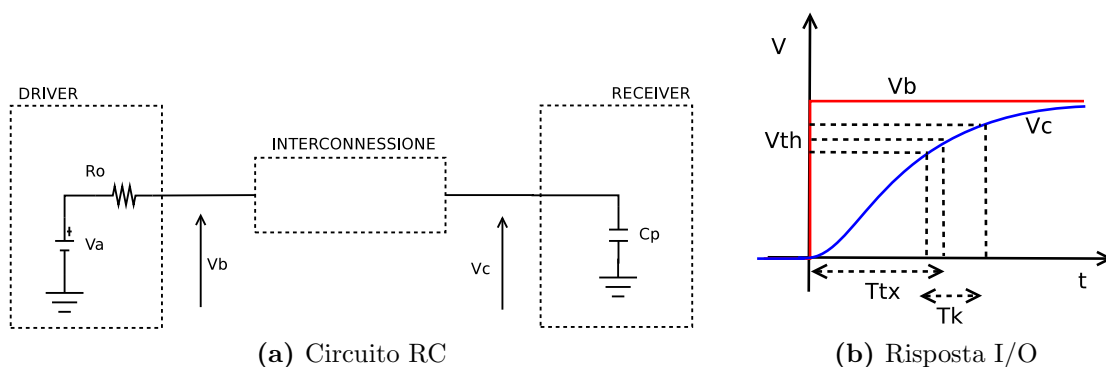
#### 6.1.1 Interconnessione ideale

Un'interconnessione ideale massimizza la velocità di trasmissione e minimizza il numero di errori

### 6.2 Modello RC passa basso o lineare

Il modello RC è un modello che descrive le entità che intervengono nell'interconnessione tra due circuiti attraverso resistenze e capacità:

- il driver è modellizzato da circuito di Thevenin
- il receiver è modellizzato da una capacità
- l'interconnessione è modellizzata da un cavo senza perdite di corrente: i punti ai capi del cavo sono equipotenziali



**Figura 6.1:** Modello lineare

## 6.2.1 Ritardi

### 6.2.1.1 Tempo di trasmissione

Il tempo di trasmissione  $t_{TX}$  è l'intervallo di tempo medio che intercorre tra l'invio di uno stato logico da parte del driver e il riconoscimento dello stato logico da parte del receiver

In un modello lineare il tempo di trasmissione dipende dai valori (variabili in certi intervalli) assunti dalle seguenti grandezze:

- livelli iniziale e finale di uscita del driver
- soglia del receiver
- resistenza di uscita del driver
- capacità di ingresso del receiver

### 6.2.1.2 Skew

Lo skew (o disallineamento) è la differenza tra il tempo di trasmissione massimo e il tempo di trasmissione minimo

$$t_k = t_{TX_{max}} - t_{TX_{min}} \quad (6.1)$$

Lo skew riduce il tempo di setup minimo del ricevitore

## 6.3 Modello a parametri concentrati

Il modello a parametri concentrati è un modello che descrive le entità che intervengono nell'interconnessione tra due circuiti attraverso resistenze, induttanze e capacità:

- il driver è modellizzato da circuito di Thevenin
- il receiver è modellizzato da una capacità
- l'interconnessione è modellizzata da una serie di celle composte da induttanze e capacità

Il modello a parametri concentrati funziona bene per i circuiti stampati

## 6.4 Modello a linea di trasmissione

Il modello a parametri concentrati è un modello che descrive le entità che intervengono nell'interconnessione tra due circuiti attraverso una semplificazione del modello a parametri concentrati; i parametri del modello a linea di trasmissione sono:

- il driver è modellizzato da circuito di Thevenin
- il receiver è modellizzato da una capacità
- l'interconnessione è modellizzata da una linea di trasmissione; i parametri di una linea di trasmissione sono:

– capacità  $C_U$  e induttanza  $L_U$  unitarie

\* per piste strette

- aumenta  $L \implies$  aumenta  $Z_\infty$
- aumenta  $L \implies$  diminuisce  $P$

- \* per piste larghe
  - diminuisce  $L \implies$  diminuisce  $Z_\infty$
  - aumenta  $C \implies$  diminuisce  $P$
- impedenza caratteristica

$$Z_\infty = \sqrt{\frac{L_U}{C_U}} \sim (10, 1000)\Omega \quad (6.2)$$

- velocità di propagazione

$$P = \frac{1}{\sqrt{L_U C_U}} \sim (0.6, 0.8)c \sim (18, 24)cm/ns \quad (6.3)$$

- lunghezza  $l$
- tempo di propagazione  $t_P$  (dipende solo da come è fatta la linea di trasmissione)
- tempo di trasmissione  $t_{TX}$  e skew  $t_k$  (dipendono dalle soglie di driver e receiver e da  $t_P$ )

A regime la linea di trasmissione è un collegamento diretto

### 6.4.1 Esempi di linee di trasmissione

Esempi di linee di trasmissione sono:

- cavi coassiali ( $Z_\infty \sim (47, 100)\Omega$ )
- cavi piatti ( $Z_\infty \sim (100, 1000)\Omega$ )
- doppiini ( $Z_\infty \sim (100, 600)\Omega$ )
- piste su circuito stampato ( $Z_\infty \sim (10, 300)\Omega$ )

### 6.4.2 Onda incidente

L'onda incidente o prima onda  $V_B(t)$  è l'onda che contiene il segnale inviato dal driver

$$V_B(t) = \frac{Z_\infty}{Z_o + Z_\infty} V_A \quad (6.4)$$

### 6.4.3 Onda di riflessione

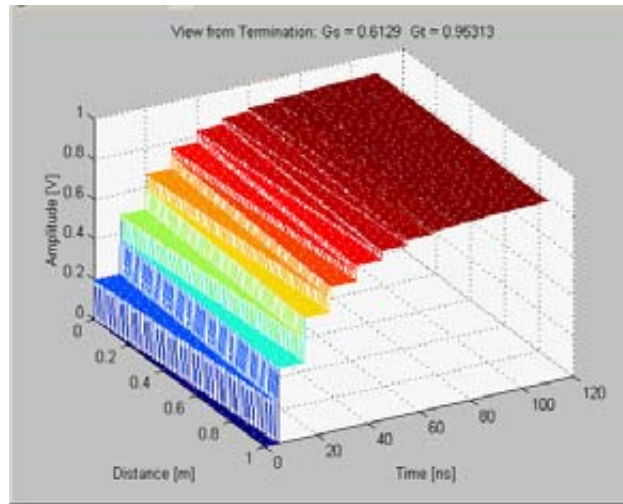
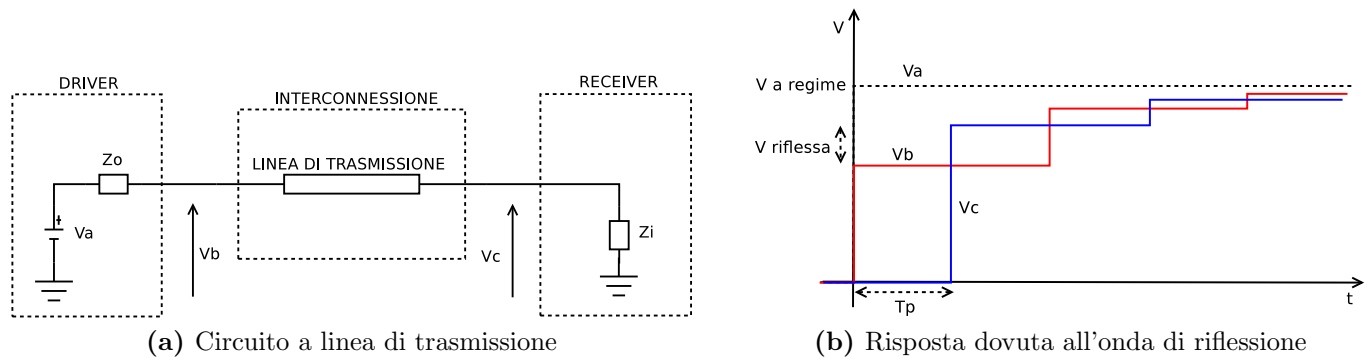
L'onda di riflessione è un segnale che “rimbalza” e torna verso il driver se: l'impedenza caratteristica  $Z_\infty$  della linea di trasmissione varia da in un certo punto della linea; oppure se l'impedenza equivalente del driver  $Z_o$  o del receiver  $Z_i$  sono diverse da  $Z_\infty$

$$V_r(t) = \Gamma V_B(t) \quad (6.5)$$

#### 6.4.3.1 Coefficiente di riflessione

Il coefficiente di riflessione misura la quantità di corrente riflessa dall'onda di riflessione

$$\Gamma = \frac{Z - Z_\infty}{Z + Z_\infty} = \begin{cases} 0 & \iff Z = Z_\infty \text{ (linea chiusa)} \\ 1 & \iff Z \rightarrow \infty \text{ (linea aperta)} \\ -1 & \iff Z = 0 \text{ (linea in corto)} \end{cases} \quad (6.6)$$



(c) Diagramma tensione-spazio-tempo

**Figura 6.2:** Modello a linea di trasmissione

### 6.4.3.2 Tensione sulla linea

La tensione totale sulla linea di trasmissione è la somma di onda incidente e di onda riflessa

## 6.5 Topologie di connessione

Le connessioni possono essere classificate a seconda della loro topologia in:

- connessioni punto-punto (un driver, una linea di trasmissione, almeno un receiver e una terminazione)
- connessioni a bus (multi driver e multi receiver)

### 6.5.1 Prestazioni

Le prestazioni di una connessione dipendono dal rapporto tra  $Z_o$  e  $Z_\infty$ :

- commutazione su onda incidente (Incident Wave Switching):  $Z_o \ll Z_\infty$ 
  - vantaggi: la soglia  $V_{th}$  viene attraversata al primo gradino: quindi alta velocità; consente il pilotaggio delle uscite in parallelo
  - svantaggi: il driver ha  $\Gamma < 0$ : quindi per evitare onde riflesse negative è necessario che la resistenza del receiver  $Z_R = Z_\infty$ : quindi la resistenza  $Z_R$  scarica corrente a massa: quindi c'è un consumo di energia elevato



- commutazione su onda riflessa (prima riflessione):  $Z_o = Z_\infty$ 
  - vantaggi: la terminazione  $Z_R$  è un circuito aperto: quindi basso consumo di energia
  - svantaggi: la soglia  $V_{th}$  può non essere attraversata al primo gradino: il ritardo massimo del sistema si ha nel lato driver con  $t_k = 2t_p$ ; consente il pilotaggio delle uscite da parte di un solo driver
- commutazione su riflessioni multiple:  $Z_o \gg Z_\infty$ 
  - vantaggi: basso consumo di energia
  - svantaggi: le salite sono molto lente dovute a riflessioni multiple

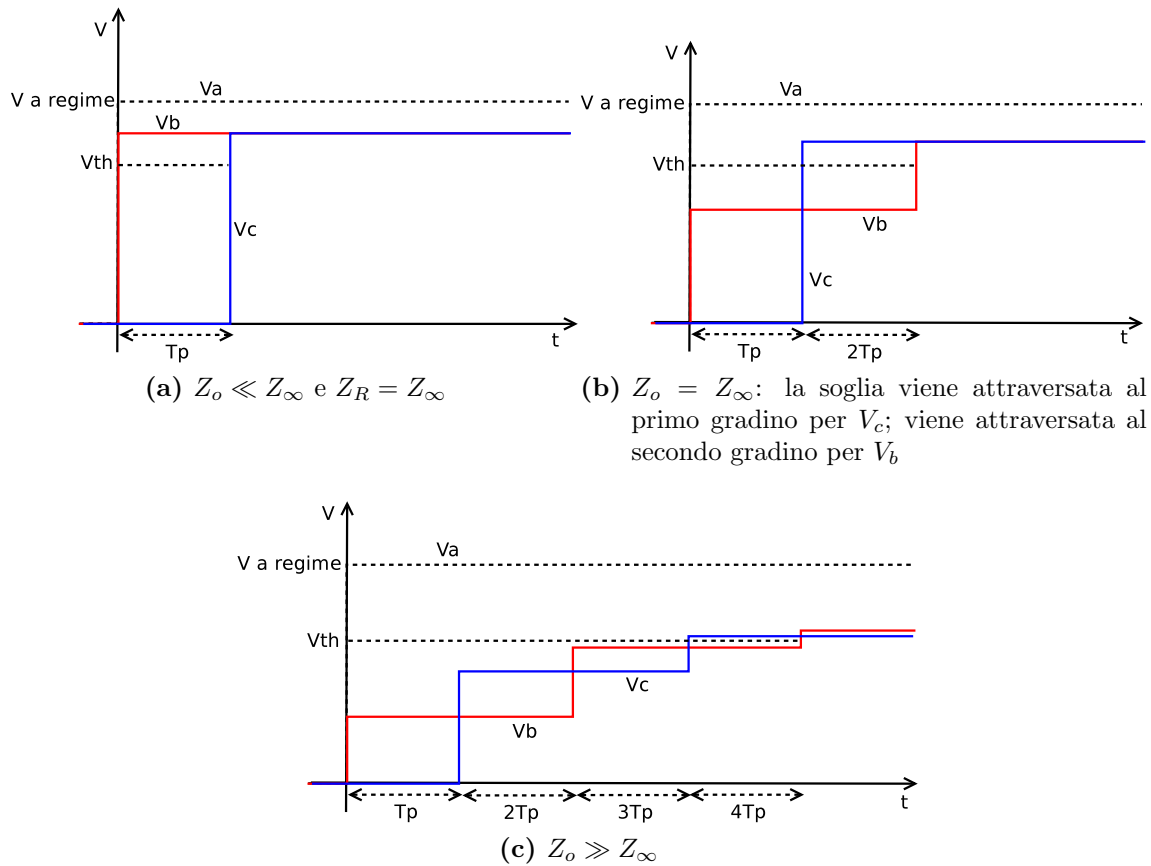


Figura 6.3: Prestazioni



# Capitolo 7

## Cicli di trasferimento

### 7.1 Cicli di trasferimento

#### 7.1.1 Modello Sor-Dest

Il modello Sor-Dest è composto da:

- sorgente
- destinazione
- $n$  connessioni per il trasporto di singoli bit
- connessioni per il controllo

#### 7.1.2 Ciclo completo

Un ciclo completo o chiuso è un ciclo di trasferimento in cui lo stato iniziale e finale sono uguali

#### 7.1.3 Classificazione dei cicli di trasferimento

I cicli di trasferimento delle informazioni sono classificati in:

- ciclo di scrittura: i bit di controllo e di informazione viaggiano nella stessa direzione; attivato dalla sorgente
  - parametro critico: skew  $t_k$
  - veloci (le memorie DDR quando devono essere lette effettuano una scrittura nei registri della CPU per far viaggiare dati e segnali di controllo nella stessa direzione e aumentare la velocità)
- ciclo di lettura: i bit di controllo e di informazione viaggiano in direzioni opposte; richiesto dalla destinazione
  - parametro critico: tempo di trasmissione  $T_{TX_{max}}$
  - lente

#### 7.1.4 Protocolli base

I protocolli base per i cicli di trasferimento sono:

- protocollo sincrono
  - temporizzazione fissa delle operazioni

- garantito il rispetto delle specifiche del caso peggiore per ogni operazione
- protocollo asincrono
  - temporizzazione adattiva delle operazioni
  - prima di continuare ogni modulo attende un ACK (ACKnowledge: conferma) dell'altro modulo coinvolto nel trasferimento

## 7.2 Protocolli per cicli di trasferimento

### 7.2.1 Protocollo sincrono

Un protocollo sincrono è un protocollo in cui la sorgente inserisce i ritardi fissi:

- $t_A$ : per garantire il tempo di setup  $t_s$  alla destinazione
- $t_B$ : per garantire il tempo di hold  $t_h$  alla destinazione

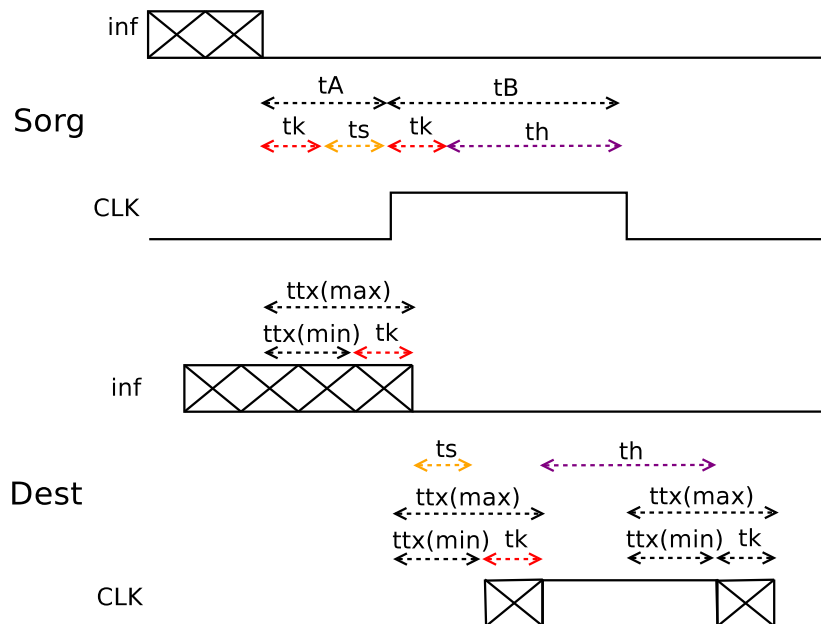
$$\begin{cases} t_A \geq t_s + t_k \\ t_B \geq t_h + t_k \end{cases} \implies t_{WR} \geq 2t_k + t_s + t_h \quad (7.1)$$

$$t_{RD} \geq t_a + t_s + t_h + 4t_{TX_{max}} \quad (7.2)$$

I tempi  $t_s$  e  $t_h$  sono fissi e dipendono dal FF, mentre lo skew  $t_k$  è variabile e dipende dall'interconnessione

Il ritardo  $t_a$  è interno alla destinazione

La sorgente deve conoscere i parametri di temporizzazione della destinazione



**Figura 7.1:** Tempo di scrittura

### 7.2.2 Protocollo asincrono

Un protocollo asincrono è un protocollo con handshake (interlacciamento con conferme) in cui:

- la sorgente invia un'informazione alla destinazione
- dopo un tempo  $t_{TX_{max}}$  la destinazione acquisisce il segnale
- un circuito tra sorgente e destinazione inserisce i tempi di setup  $t_s$  e hold  $t_h$  per la destinazione
- il circuito frapposto invia un ACK alla sorgente
- dopo un tempo  $t_{TX_{max}}$  la sorgente acquisisce l'ACK e invia un ACK di conferma alla destinazione
- dopo un tempo  $t_{TX_{max}}$  la destinazione acquisisce l'ACK di conferma e invia un ACK alla sorgente
- dopo un tempo  $t_{TX_{max}}$  la sorgente acquisisce l'ultimo ACK

$$t_{WR} \geq t_k + t_s + t_h + 4t_{TX_{max}} \quad (7.3)$$

$$t_{RD} \geq t_k + t_1 + t_2 + 4t_{TX_{max}} \quad (7.4)$$

Il ritardo  $t_1$  è dovuto al setup e all'hold della destinazione; il ritardo  $t_2$  è il tempo di accesso alla sorgente dati. La sorgente non deve conoscere a priori le tempistiche del modulo destinazione.



# Capitolo 8

## Collegamenti paralleli

### 8.1 Protocolli di un sistema a BUS

I protocolli di un sistema a BUS sono:

- protocollo di allocazione: serve per selezionare correttamente il Master del BUS
- protocollo di indirizzamento: serve per selezionare correttamente uno Slave
- protocollo di trasferimento: serve per trasferire correttamente le informazioni tra Master e Slave

### 8.2 Allocazione

#### 8.2.1 Tecniche di gestione del bus

Le tecniche di gestione del bus sono:

- token passing: il GRANT viene assegnato ai Master a turno
  - svantaggio: il GRANT viene assegnato anche ai Master che non ne hanno bisogno
- collision detection: il GRANT viene assegnato a tutti i Master che lo richiedono
  - svantaggio: perdite di tempo in caso di collisioni (accessi contemporanei al bus da parte di due Master)
- arbitraggio: un arbitro riceve le REQUEST dei Master e assegna ad un solo Master alla volta il GRANT

#### 8.2.2 Arbitraggio

L'arbitraggio del BUS si può effettuare attraverso tecniche diverse:

- First Come First Served: il primo Master a inoltrare una REQUEST ottiene il GRANT
  - svantaggio: è necessario gestire richieste contemporanee
- arbitraggio a priorità: all'interno di una finestra temporale l'arbitro immagazzina le REQUEST inoltrate e concede il GRANT in base ad una gerarchia
  - svantaggio: è necessario gestire le REQUEST a priorità più bassa perché rischiano di non essere mai servite (starvation)

- arbitraggio con fairness: l'arbitro congela lo stato delle REQUEST all'interno di una finestra temporale, serve quelle REQUEST e non ne accetta di nuove finché non ha finito di servirle
- arbitraggio a priorità rotante: ogni volta che l'arbitro concede il GRANT ad un Master la priorità dei Master cambia

## 8.3 Indirizzamento

### 8.3.1 Slave

Lo Slave è composto da 4 subunità:

- unità di decodifica di indirizzo
- unità di temporizzazione e controllo
- unità buffer dati
- unità interna (registri, memoria, unità operative)

### 8.3.2 Selezione dello Slave

La selezione dello Slave può avvenire attraverso 2 tecniche:

- indirizzamento codificato: ogni memoria è individuata da un codice binario (e.g.: 11 individua la memoria nella quarta posizione)
  - con  $N$  bit si possono raggiungere  $2^N$  dispositivi
- indirizzamento lineare: ogni memoria è individuata da un bit di una stringa di bit (e.g.: 1000 individua la memoria nella quarta posizione)
  - con  $N$  bit si possono raggiungere  $N$  dispositivi

## 8.4 Prestazioni

### 8.4.1 Parametri del BUS

I parametri del BUS sono:

- width (larghezza del BUS)  $W$
- tempo di ciclo  $t_c$ ; dipende da:
  - parametri elettrici:  $t_{TX}$  e  $t_k$
  - parametri dei moduli:  $t_s$ ,  $t_h$ ,  $t_{WR}$ ,  $t_{RD}$ ,  $t_{CLK}$ ...
  - tipo di protocollo
- speed (velocità di trasmissione del BUS):  $S$  [cicli(wr/rd)/s]

$$S = \frac{1}{t_c} \quad (8.1)$$

- throughput  $T$

$$T = W \cdot S \quad (8.2)$$



## 8.4.2 Tecniche per il miglioramento delle prestazioni

### 8.4.2.1 Source Synchronous Protocol

Un protocollo Source Synchronous è un protocollo in cui i segnali di controllo e le informazioni viaggiano sempre nella stessa direzione; le uniche operazioni svolte sono operazioni di scrittura: CPU  $\rightarrow$  memoria e memoria  $\rightarrow$  CPU

### 8.4.2.2 Multiplexed BUS

Un multiplexed BUS è un BUS in cui gli indirizzi e i dati sono multiplati nel tempo: viaggiano sullo stesso conduttore in tempi diversi (si risparmia sulla dimensione del BUS e sul consumo di energia)

### 8.4.2.3 Embedded Clock

Una modulazione embedded clock (o autosincronizzante) è una tecnica di multiplazione che permette di utilizzare lo stesso canale fisico per i segnali di temporizzazione e i segnali di informazione (indirizzi e dati) (si elimina lo skew  $t_k$ )

### 8.4.2.4 Double Data Rate (DDR) Cycle

Un ciclo DDR è un ciclo dual edge (a doppio margine): un ciclo in cui vengono usate sia le transizioni di salita sia le transizioni di discesa del segnale di STB/ACK (minor consumo e maggiore velocità)

### 8.4.2.5 Burst

Un burst (o trasferimento a blocchi) è una tecnica di trasferimento utilizzabile quando è necessario trasferire celle di memoria adiacenti:

- il master invia l'indirizzo del primo dato e il numero di dati adiacenti che intende trasferire
- il master invia i dati uno di seguito all'altro senza ripetere i loro indirizzi

Si risparmia il tempo di indirizzamento di ogni pacchetto dati

## 8.5 Limiti dei collegamenti paralleli

I limiti dei collegamenti paralleli sono:

- velocità limitata da  $T_{TX}$  e  $t_k$
- strutture multipunto richiedono terminazioni  $\implies$  dissipazione di potenza
- diafonia: il segnale su un conduttore disturba il segnale nei conduttori vicini
- per incrementare il throughput è necessario un elevato parallelismo
  - $\implies$  connettori grandi (spazio)
  - $\implies$  problemi di compatibilità
  - $\implies$  consumo elevato



# Capitolo 9

## Collegamenti seriali

### 9.1 Collegamento seriale

Un collegamento seriale è un collegamento in cui:

- il driver invia i bit relativi ad un'informazione uno di seguito all'altro
- il receiver riceve i bit sequenzialmente nell'ordine in cui il driver li ha inviati

#### 9.1.1 CLK/Data embedding

Il CLK/Data embedding (fusione di clock e dati o collegamento seriale autosincronizzante) è una tecnica di collegamento seriale in cui i dati e il segnale di temporizzazione (e.g.: CLK) sono trasportati dallo stesso conduttore

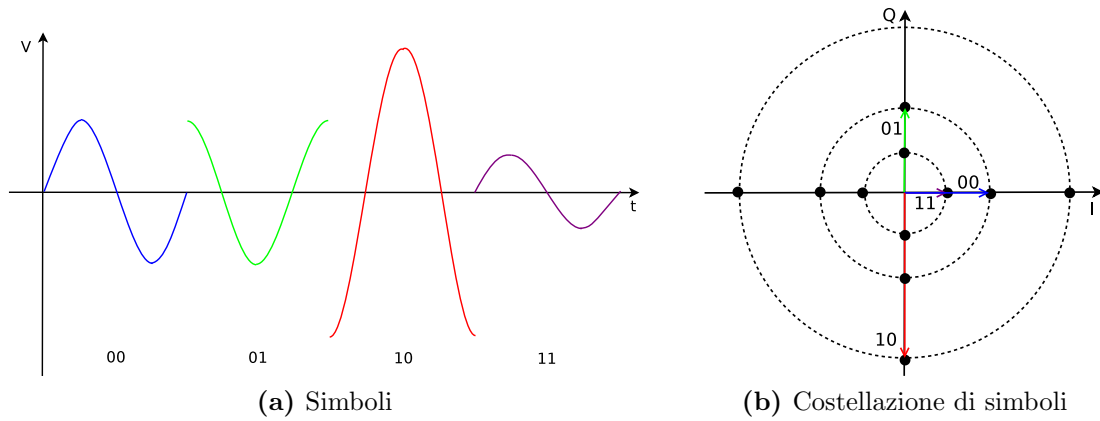
#### 9.1.2 Caratteristiche dei collegamenti seriali

Le caratteristiche dei collegamenti seriali sono:

- vantaggi
  - parametro di indeterminazione skew  $t_k$  eliminato
  - pochi conduttori
  - semplificazione routing (cablaggio)
  - riduzione del consumo (un solo driver)
  - alta efficienza per percorsi lunghi e/o ad alta velocità
- svantaggi
  - ritardi nel trasferimento di più bit (per ogni ciclo viene trasportato un solo bit)
  - clock fuso con i dati

### 9.2 Simbolo

Un simbolo è la più piccola quantità di bit che può essere trasmessa in una sola volta in un canale di trasmissione



**Figura 9.1:** Rappresentazione dei simboli

### 9.2.1 Costellazione di segnali

Una costellazione di segnali è una rappresentazione di un segnale tramite circonferenze goniometriche

## 9.3 Parametri dei collegamenti

### 9.3.1 Bit rate

Il bit rate  $R$  è la quantità di dati che possono essere trasferiti da una connessione in un intervallo di tempo

$$R = \frac{n_{bit}}{\Delta t} \left[ \frac{bit}{s} \right] \quad (9.1)$$

### 9.3.2 Baud rate

Il baud rate (o symbol rate)  $SR$  è la quantità di simboli che possono essere trasferiti da una connessione in un intervallo di tempo

$$SR = \frac{n_{symbol}}{\Delta t} \left[ baud = \frac{symbol}{s} \right] \quad (9.2)$$

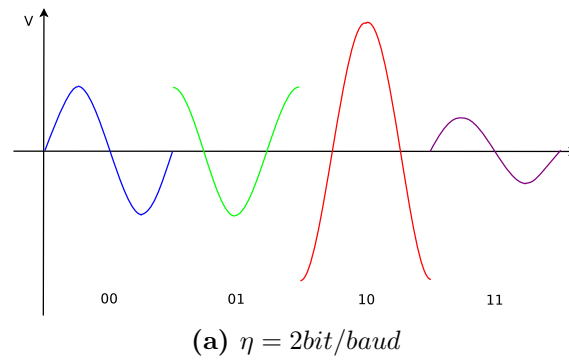
### 9.3.3 Efficienza

L'efficienza  $\eta$  è la quantità di bit che possono essere trasferiti da una connessione in un intervallo di tempo

$$\eta = \frac{n_{bit}}{n_{symbol}} \left[ \frac{bit}{baud} \right] \quad (9.3)$$

### 9.3.4 ISI

L'ISI (Inter Symbolic Interference) è un'interferenza provocata dal trasferimento di segnale da un intervallo temporale ad un altro in uno stesso conduttore (come un'eco provocata dal segnale all'interno del conduttore)



**Figura 9.2:** Symbol rate e efficienza

### 9.3.5 Sincronismo

Le tipologie di sincronismo sono:

- sincronismo di bit: garantisce il corretto campionamento del singolo bit; è legato alle relazioni di fase tra transizioni dati e il clock del receiver
- sincronismo di carattere: garantisce il corretto riconoscimento del primo e dell'ultimo bit di una stringa; è legato all'attivazione del segnale READY

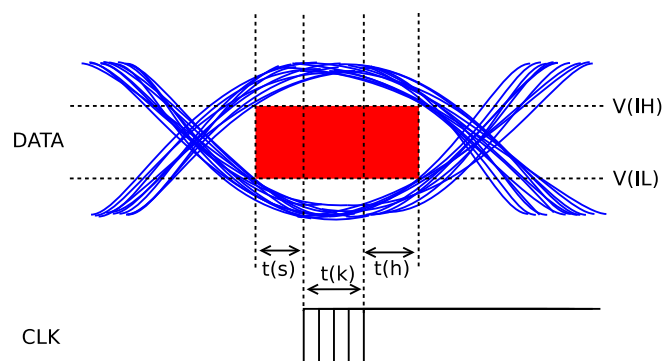
## 9.4 Diagramma a occhio

Un diagramma a occhio è un diagramma che rappresenta l'intervallo di tempo in cui un segnale può essere campionato correttamente perché il costruttore garantisce che il segnale inviato dal driver sia costante

### 9.4.1 Parametri di un diagramma ad occhio

I parametri di un diagramma ad occhio sono:

- eye opening: zona di corretto campionamento
  - $V \in (V_{IL}, V_{IH})$
  - $t \geq t_s + t_h$
  - indeterminazione del clock  $t_k$



**Figura 9.3:** Diagramma ad occhio

## 9.5 Collegamenti seriali

### 9.5.1 Collegamenti seriali sincroni e asincroni

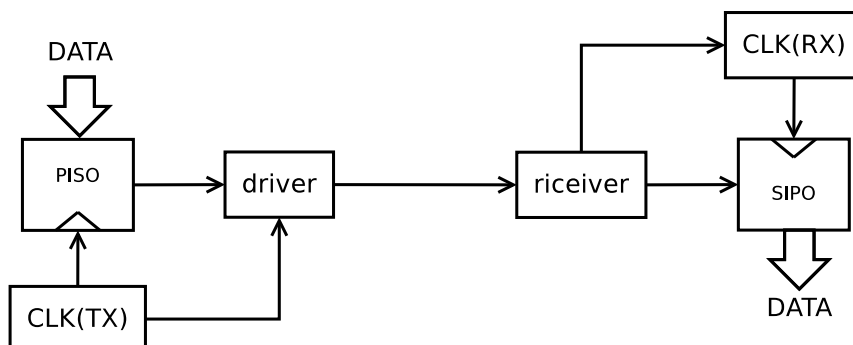
I collegamenti seriali possono essere:

- sincroni: i bit sono organizzati in gruppi detti caratteri
  - la trasmissione di bit è discontinua
  - il CLK viene sincronizzato dallo start bit e dallo stop bit che indicano l'inizio e la fine di un carattere
- asincroni: i bit sono organizzati in pacchetti
  - la trasmissione di bit è continua
  - il CLK si deve sincronizzare su ogni bit

### 9.5.2 Tecniche di sincronizzazione

Le tecniche di sincronizzazione per i collegamenti seriali sono:

- singolo clock comandato dal driver su conduttore separato
  - funzionamento: il driver genera un segnale di clock che viaggia su un conduttore separato dai dati e comanda il clock del receiver
  - vantaggi: massima velocità
  - svantaggi: indeterminazione (skew)
- singolo clock comandato dal receiver su conduttore separato
  - funzionamento: il receiver genera un segnale di clock che viaggia su un conduttore separato dai dati e comanda il clock del driver
  - vantaggi: il receiver può controllare la velocità di comunicazione
  - svantaggi: ritardi maggiori
- clock embedded
  - funzionamento: clock e dati viaggiano su un unico conduttore
  - vantaggi: si elimina il parametro di indeterminazione (skew)
  - svantaggi: sono necessari protocolli aggiuntivi per fondere e separare i dati dal clock

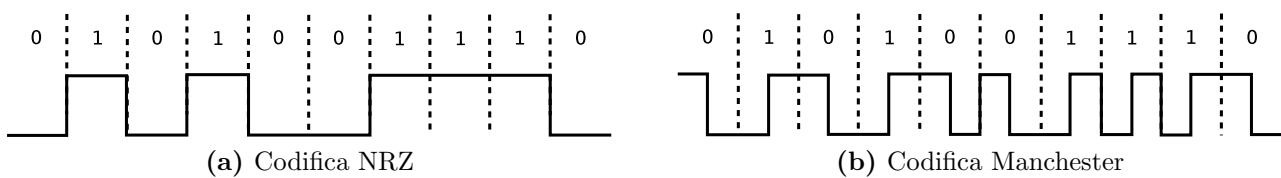


**Figura 9.4:** Clock embedding

### 9.5.3 Codifiche seriali

La codifiche dei collegamenti seriali sono:

- codifica NRZ (Non Return to Zero)
  - funzionamento: 1 è rappresentato da uno stato (H o L); 0 si comporta all'opposto
  - banda:  $B = 1$  transizione/bit
  - frequenza:  $f_{max} = R/2$  bit/s
  - svantaggio: per lunghe sequenze di bit costanti (0 o 1) si rischia di perdere il sincronismo
- codifica Manchester
  - funzionamento: 1 è rappresentato da una transizione (H→L o L→H); 0 si comporta all'opposto
  - banda:  $B = 2$  transizioni/bit
  - frequenza:  $f_{max} = R$  bit/s
  - vantaggi: permette di sincronizzare i clock ad ogni bit; permette il clock embedded

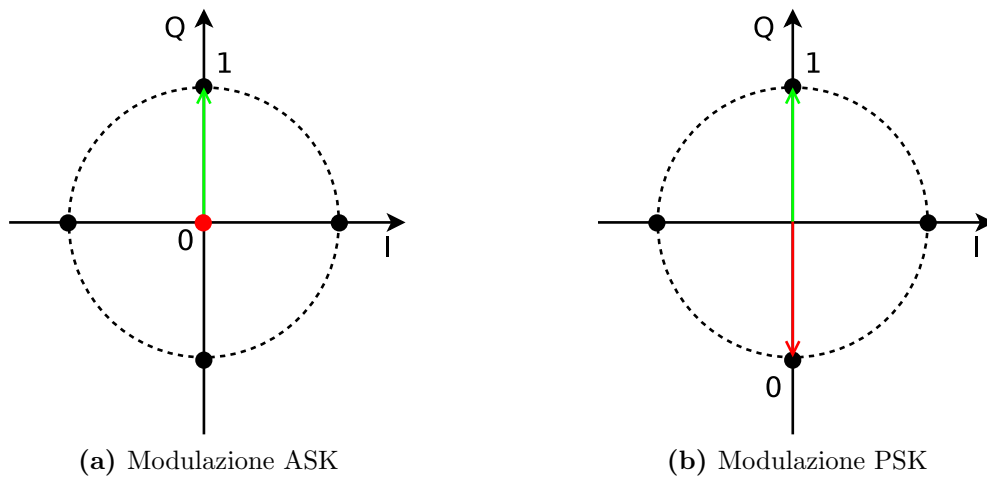


**Figura 9.5:** Codifiche seriali

## 9.6 Modulazione dei segnali analogici

Le modulazioni dei segnali analogici possono essere:

- ASK (Amplitude Shift Keyed)
  - funzionamento: 1: la portante viene moltiplicata per 1; 0: la portante viene moltiplicata per 0
- PSK (Phase Shift Keyed)
  - funzionamento: 1: la portante viene sfasata di  $\pi$ ; 0: la portante viene sfasata di  $\pi$



**Figura 9.6:** Modulazioni ASK e PSK



# Capitolo 10

## Integrità di segnale

### 10.1 Diafonia

La diafonia è un disturbo causato dal passaggio di segnale tra due canali che dovrebbero essere separati:

- nello spazio: è dovuta all'accoppiamento di due conduttori tramite induttanze e capacità
- nel tempo (Inter Symbolic Interference): è dovuta alle “code” dei segnali

#### 10.1.1 Tipologie

La diafonia è:

- diretta: quando il segnale (onda incidente) e il disturbo (rumore incidente) viaggiano nella stessa direzione
  - effetto: i disturbi generati nel tempo si sommano in ampiezza
- inversa: quando il segnale (onda incidente) e il disturbo (rumore riflesso) viaggiano in direzioni opposte
  - effetto: i disturbi generati nel tempo si sommano in ampiezza

#### 10.1.2 Soluzioni

Le soluzioni alla diafonia sono:

- riduzione della diafonia all'origine
  - rallentare dei fronti ( $dV/dt$ ) del driver della linea disturbante per evitare di inserire componenti ad alta frequenza
    - \* ridurre  $dV$  (segnali a bassa energia)
    - \* aumentare  $dt$
    - \* vantaggio: riduzione dei consumi
    - \* svantaggio: diminuisce la velocità di trasmissione
  - ridurre capacità e induttanze mutue
  - usare segnali differenziali
    - \* vantaggi: immunità al rumore, riduzione del consumo
    - \* svantaggi: piste doppie per ogni segnale, richiede tecniche analogiche
- riduzione degli effetti della diafonia

- filtrare i receiver della linea disturbata
- tecniche EDC

## 10.2 Rumore di commutazione

Il rumore di commutazione è un disturbo causato dalle scariche di corrente delle induttanze parassite durante le commutazioni  $H \rightarrow L$  e  $L \rightarrow H$ :

- ground bounce (rimbalzo di massa)
  - effetto: può trasformare uno 0 logico in 1
- power bounce (rimbalzo di alimentazione)
  - effetto: può trasformare un 1 logico in 0

### 10.2.1 Soluzioni

Le soluzioni al rumore di commutazione sono:

- condensatori bypass
  - funzionamento: le correnti impulsive vengono fornite dalle scariche di condensatori in parallelo ai receiver

# Parte III

## Conversioni A/D/A



# Capitolo 11

## Introduzione

### 11.1 Segnali analogici e digitali

Le definizioni di segnale analogico e segnale digitale sono:

- segnale analogico: segnale continuo la cui variazione nel tempo è una rappresentazione di un'altra grandezza
- segnale digitale: segnale discreto nel tempo e in ampiezza

#### 11.1.1 Comunicazione A/D/A

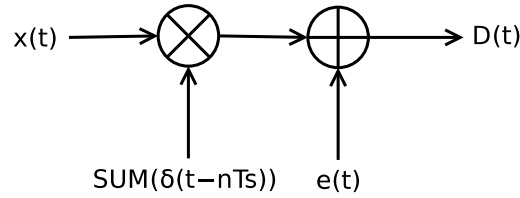
Il mondo digitale e il mondo analogico comunicano nel seguente modo:

- sensori: trasformano una grandezza naturale (temperatura, pressione, ...) in un segnale analogo (con caratteristiche che possano essere ricondotte alla grandezza), ma di altra natura (tensione, corrente, ...)
- protettore: protegge il sistema da correnti e tensioni eccessive
- amplificatore: amplifica il segnale per portarlo vicino al fondo scala (per migliorare il SNR)
- filtro anti-alias: filtro passa basso per eliminare le alte frequenze
- multiplexer
- sampler: campiona il segnale in ingresso
- holder: mantiene il segnale campionato costante per permettere al sistema di avere il tempo di elaborarlo
- convertitore analogico/digitale: trasforma i segnali analogici (tensione, corrente, ...) in segnali digitali (bit)
- unità di elaborazione: processa i segnali digitali
- convertitore digitale/analogico: trasforma i segnali digitali (bit) in segnali analogici (tensione, corrente, ...)
- filtro di ricostruzione: per eliminare errori e rumori
- amplificatore
- attuatori: trasformano un segnale analogico (tensione, corrente, ...) in un'azione sul mondo naturale (movimenti, ...)

## 11.2 Conversione A/D

La conversione A/D viene attuata in due passaggi:

- campionamento: il segnale analogico viene discretizzato nel tempo
- quantizzazione: il segnale campionato viene discretizzato in ampiezza



**Figura 11.1:** Conversione A/D

### 11.2.1 Campionamento

Il campionamento di un segnale viene effettuato moltiplicando il segnale per un treno di delta di Dirac

$$x_s(t) = x(t) \sum \delta(t - nT_s) \quad (11.1)$$

#### 11.2.1.1 Alias

Gli alias sono delle repliche in frequenza dello spettro principale di un segnale che si formano quando il segnale viene campionato

#### 11.2.1.2 Aliasing

L'aliasing è il fenomeno per cui lo spettro principale di un segnale si sovrappone a quello dei suoi alias

#### 11.2.1.3 Filtro anti aliasing

Un filtro anti aliasing è un filtro che si applica ad un segnale analogico prima di campionarlo per limitare la sua banda a  $f_s/2$  ed evitare che si generi l'aliasing



**Figura 11.2:** Filtro anti aliasing

#### 11.2.1.4 Rumore di alising

Il rumore di aliasing è un rumore residuo dovuto agli alias che il filtro non è riuscito ad eliminare

#### 11.2.1.5 Teorema di Nyquist-Shannon

La frequenza di campionamento di un segnale deve essere sempre almeno il doppio della banda dello spettro principale del segnale da campionare

$$f_s \geq 2B \quad (11.2)$$

### 11.2.1.6 Hold

L'hold (mantenimento) è una tecnica che permette di mantenere per un certo intervallo di tempo un segnale campionato per aver tempo di effettuare operazioni sul campione

Gli effetti dell'hold sono:

- nel dominio del tempo: trasforma impulsi in gradini
- nel dominio della frequenza: attenua le componenti a frequenza più elevata

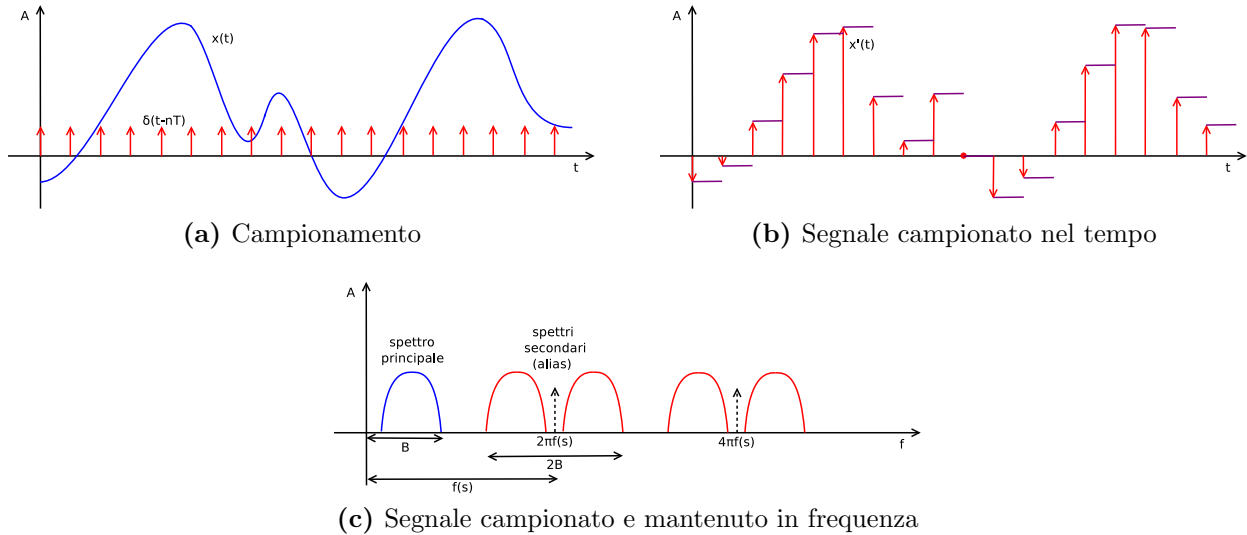


Figura 11.3: Campionamento

## 11.2.2 Quantizzazione

La quantizzazione di un segnale campionato viene effettuata:

- dividendo l'asse delle ampiezze in  $N$  intervalli di dimensione  $A_d$
- assegnando a ciascun intervallo un valore numerico ( $2^N$  valori)

### 11.2.2.1 Errore di quantizzazione

L'errore di quantizzazione è un errore dovuto all'impossibilità di ricostruire i valori analogici esatti all'interno di un intervallo  $\epsilon_q$ ; il massimo scostamento rispetto al valore centrale dell'intervallo è ( $S$ ):

$$|\epsilon_q| \leq \frac{S}{2^{N+1}} \quad (11.3)$$

L'errore di quantizzazione è inversamente proporzionale al numero di livelli (bit) utilizzati per rappresentare il segnale

### 11.2.2.2 Rapporto segnale/rumore (SNR)

Il SNR (Signal/Noise Ratio) è un indice della qualità del segnale: è il rapporto tra la potenza media del segnale e la potenza media del rumore

$$SNR = \frac{P_s}{P_r} \quad (11.4)$$

Le proprietà del SNR sono:

- per segnali che arrivano al fondo scala dipende dal numero di bit utilizzati nella quantizzazione ( $6N$  dove  $N$  è il numero di bit) e dalla forma d'onda ( $k$ )

$$SNR_q = 6N + k[dB] \quad (11.5)$$

- per segnali che non arrivano al fondo scala diminuisce (peggiora) perché diminuisce la potenza del segnale mentre la potenza del rumore rimane costante (dipende solo dal numero di bit)

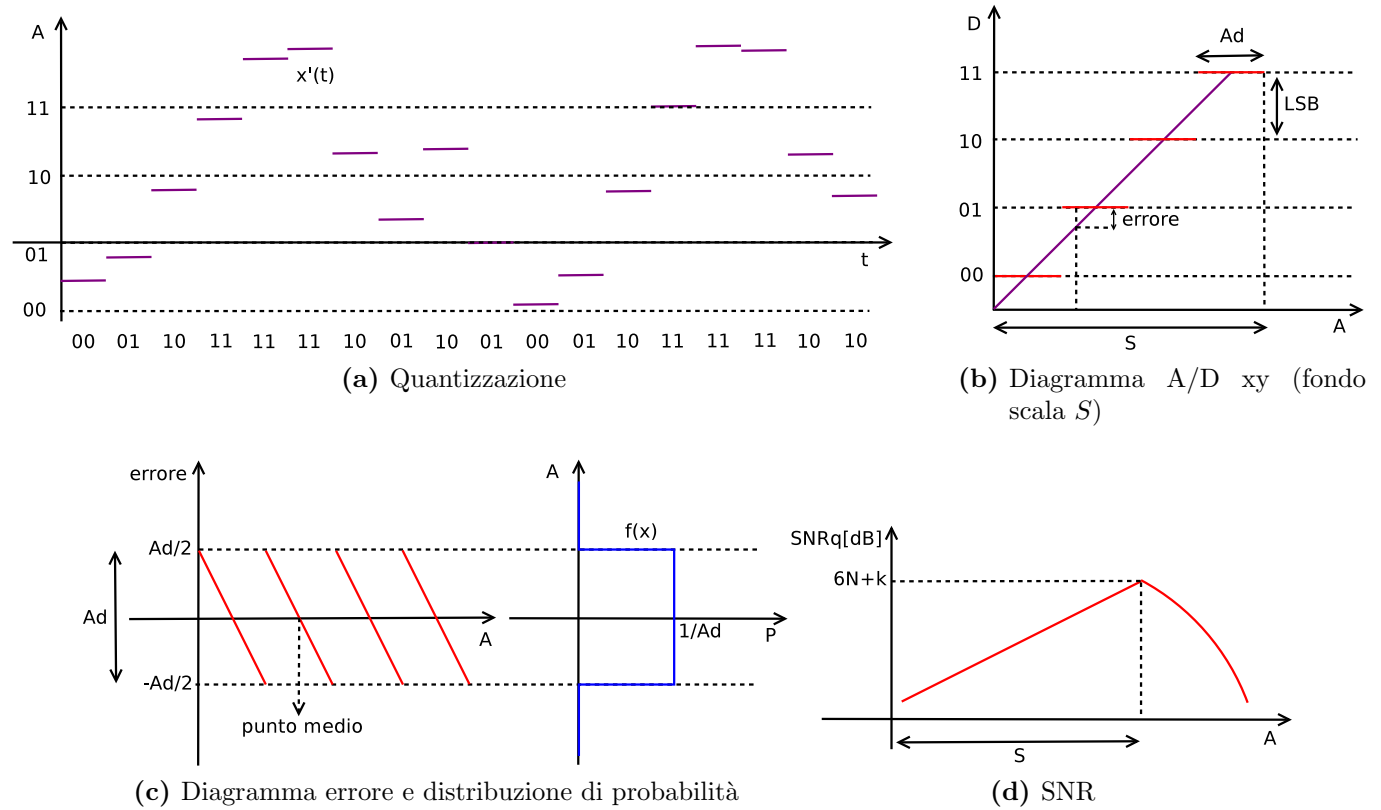
### 11.2.2.3 Distribuzione di probabilità

La distribuzione di probabilità  $f(x)$  del segnale e del rumore è un parametro utile a descrivere il generico comportamento di un sistema

### 11.2.2.4 Potenza dell'errore

La potenza dell'errore di quantizzazione è definita come

$$\sigma_{\epsilon_q}^2 = \int_{-A_d/2}^{A_d/2} \epsilon_q^2 f(\epsilon_q) d\epsilon_q \quad (11.6)$$



**Figura 11.4:** Quantizzazione

### 11.2.2.5 ENOB

L'ENOB (Effective Number Of Bits) è il parametro che rappresenta il numero di bit significativi per un sistema di conversione: tiene conto dell'errore totale di tutto il sistema di conversione

$$ENOB = \frac{SNR_{tot} - k}{6} < N \quad (11.7)$$



# Capitolo 12

## Convertitori D/A

### 12.1 Parametri

#### 12.1.1 Caratteristica di conversione D/A

La caratteristica ideale di ingresso  $D$  di una convertitore D/A è una sequenza di  $M = 2^N$  (dove  $N$  è il numero di bit) punti equispaziati ed allineati

$$A = KD \quad (12.1)$$

La caratteristica reale di ingresso di un convertitore D/A è una sequenza di punti curvilinea compresa nella fascia di non linearità

$$A = K'D(\epsilon_{nli}; \epsilon_{nld}) + V_{offset} \quad (12.2)$$

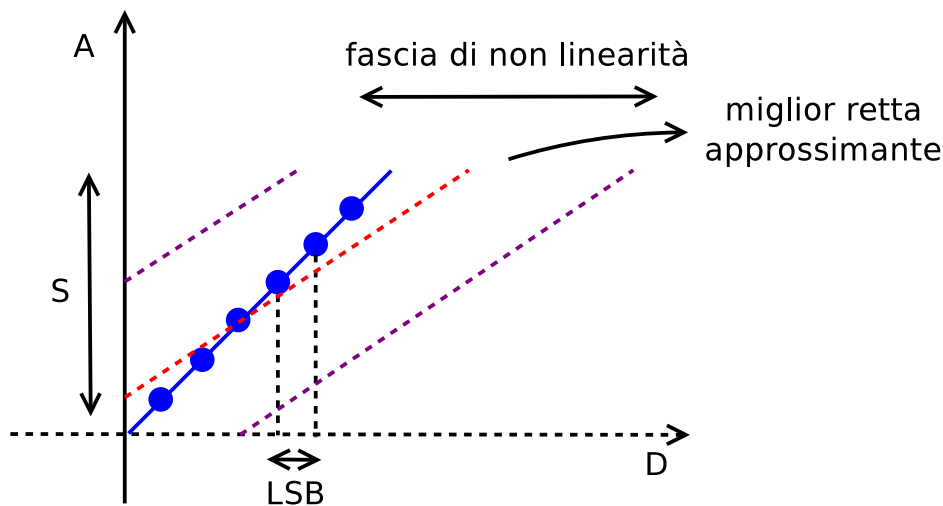


Figura 12.1: Caratteristica di conversione D/A

#### 12.1.2 Parametri di un convertitore D/A

I parametri di un convertitore D/A sono:

- parametri statici
  - numero di bit (risoluzione):  $N$
  - fondo scala:  $S$
- errori

- errori lineari
  - \* errore di guadagno (pendenza)  $\epsilon_G = \frac{\Delta K}{K}$
  - \* errore di offset (traslazione)  $V_{offset}$
- errori non lineari
  - \* errore integrale (complessivo)  $\epsilon_{ni}$
  - \* errore differenziale (del singolo gradino)  $\epsilon_{nld} = A_{D_{ideale}} - A_{D_{reale}}$ 
    - errore di non monotonocità: si ha quando  $\epsilon_{nld} > 1 \text{ LSB}$  (inversione di pendenza della caratteristica)
- parametri dinamici
  - Setting Time  $t_S$  (tempo di assetto): l'intervallo di tempo necessario affinché l'uscita analogica si stabilizzi all'interno della fascia di errore di 1  $LSB$
  - glitch: fenomeno che si genera durante i transitori quando le uscite non sono ben sincronizzate: può portare l'uscita a valori molto diversi per qualche istante

## 12.2 Circuiti per convertitori D/A

### 12.2.1 Struttura dei convertitori

La struttura di un convertitore D/A può essere modellata nel seguente modo:

- input: grandezza digitale di riferimento (e.g.: tensione, corrente)
- grandezze elementari: la grandezza di riferimento viene scomposta in grandezze elementari
- interruttori: gli interruttori attivano le uscite delle grandezze elementari
- sommatore: somma le grandezze elementari attive
- output: uscita analogica

### 12.2.2 Tecniche base di progettazione

Le tecniche base di progettazione di circuiti per convertitori D/A sono:

- somma di grandezze elementari uniformi: le grandezze elementari in ingresso sono tutte uguali (e.g.: 1; per fare 9:  $1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1$ )
- somma di grandezze elementari pesate: le grandezze elementari in ingresso hanno un peso corrispondente ad una potenza di 2 (e.g.: 1, 2, 4, 8; per fare 9:  $1 + 8$ )