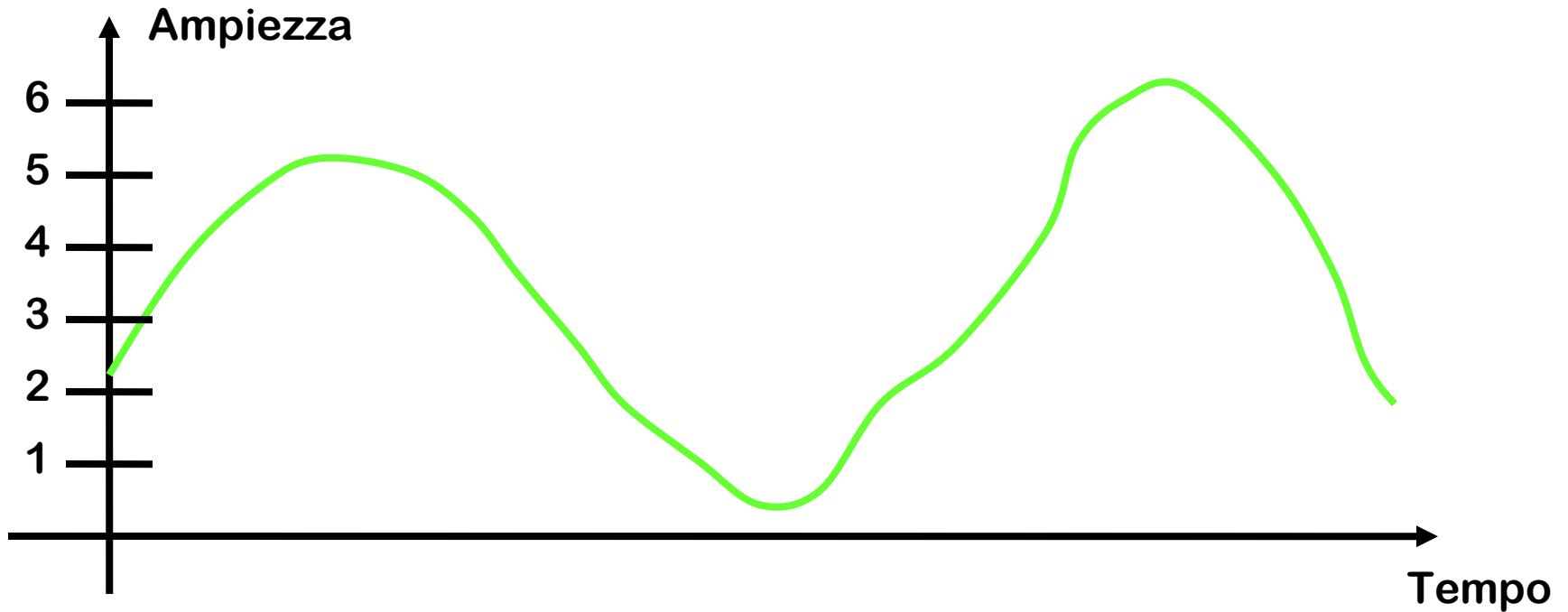


Segnali



Segnale

- Possiamo definire *segnale* la “variazione” nel tempo di una qualche grandezza fisica: es. varia l’intensità(= ampiezza).



Esempi

- Quando parliamo alteriamo la pressione dell'aria circostante. Questa **variazione di pressione** si propaga, attenuandosi progressivamente, fino all'orecchio dell'ascoltatore. Tale variazione di pressione rappresenta un **segnale “sonoro” o “acustico”**
- **Nei circuiti elettrici** quello che si propaga è una **variazione di tensione o di corrente**

Animazioni

- Altoparlante
- Sistema uditivo
- Microfono (cliccare sui vari tipi di microfono)

Segnali e codifica



Segnali e codifica

- **Attraverso le sue variazioni un segnale può trasportare informazione**, veicolando un messaggio da un mittente ad un destinatario (che sia in grado di interpretarlo...)
- **E' necessario stabilire un «protocollo» di comunicazione** in cui venga specificato il **formato** dei messaggi ovvero la loro **“codifica”**(es. lingua italiana & segnale vocale, codice gestuale(sordomuti), segnali di fumo...) e le **regole che governano la comunicazione**(es. Se parli tu, non parlo io e viceversa,...)
- **La codifica finale**, ovvero la forma in cui il segnale verrà trasmesso sul canale di comunicazione, **deve essere adatta alle caratteristiche del mezzo trasmissivo**(non posso usare la voce se l'utente si trova a 1km di distanza!)

Trasduzione

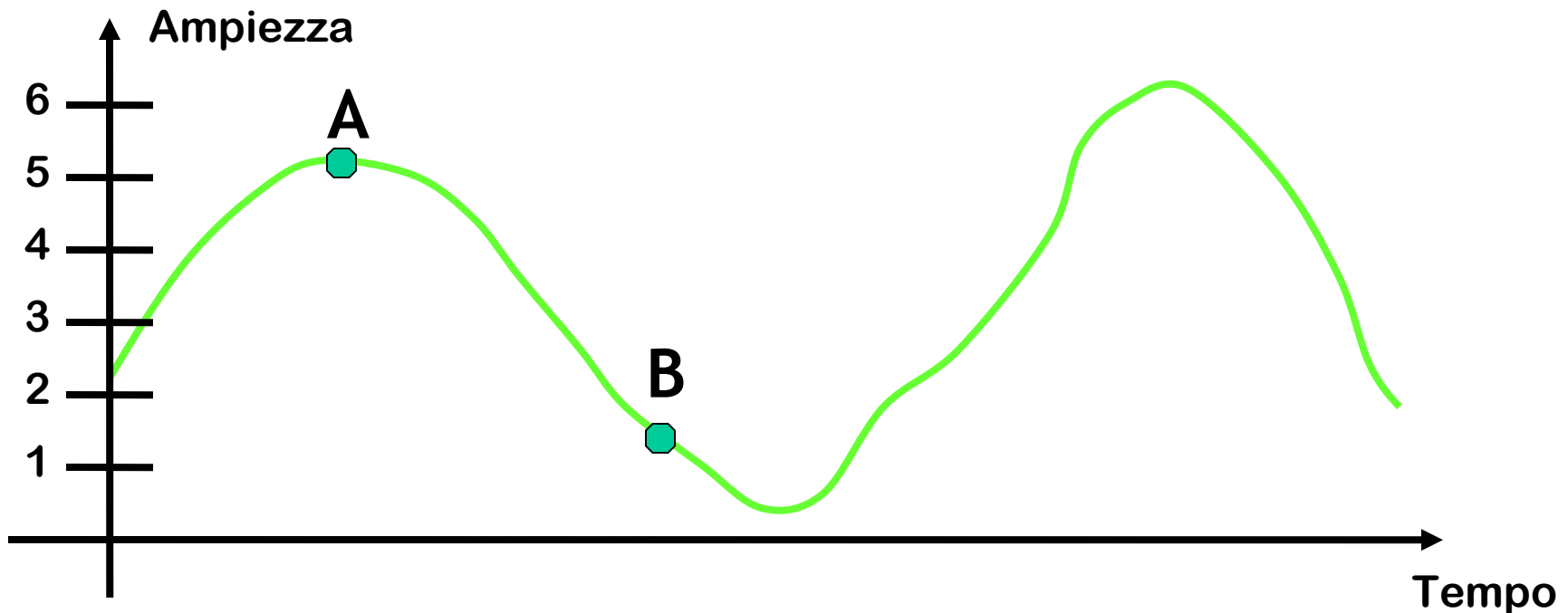
- *Quando si converte un segnale di un certo tipo, in un segnale di natura diversa si parla di **trasduzione del segnale**.*
- Il dispositivo che realizza tale conversione si chiama **trasduttore**.

Trasduttori

- *Casse dello stereo* (tensione-pressione)
- *Microfono* (pressione-tensione)
- *Termometro* (temperatura-lunghezza livello mercurio)
- *Fotocellula* (luce-tensione)
- *Lampadina* (tensione-luce)

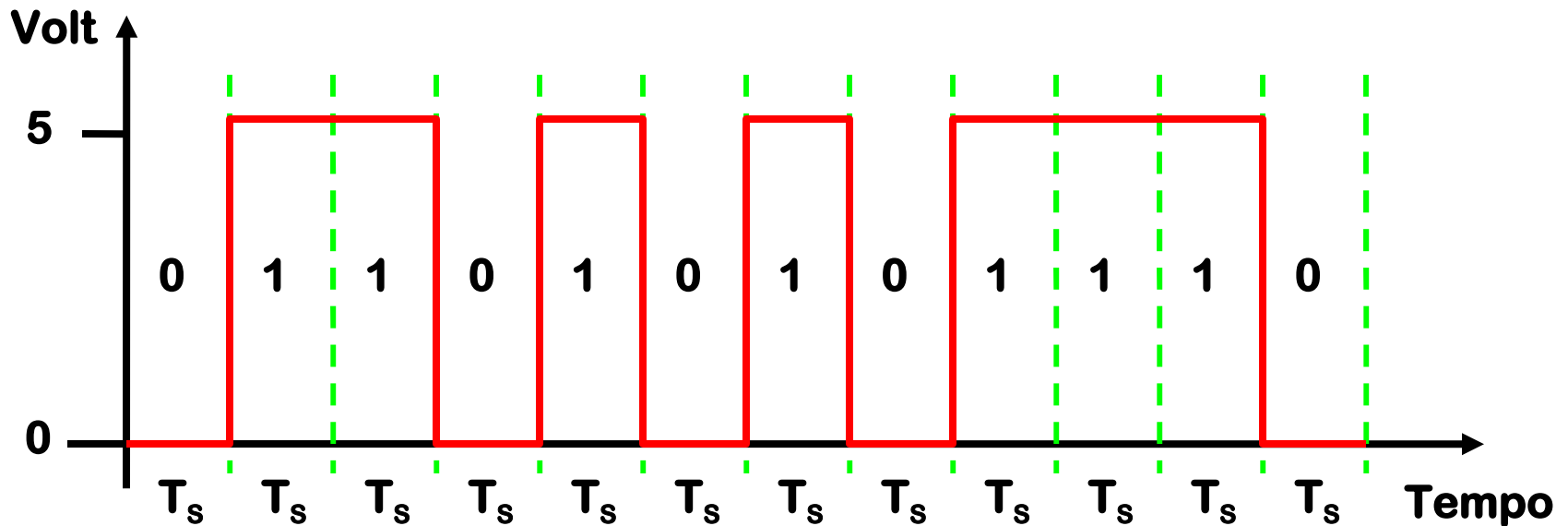
Segnale Analogico

- Per segnale *analogico* si intende un segnale la cui ampiezza **varia in maniera continua(ovvero graduale)** nel tempo.
- Passando da un valore A ad un valore B, il segnale assumerà anche tutti i valori intermedi tra A e B(una serie **infinita** di valori!!!).
- **Non vi sono salti!!!**



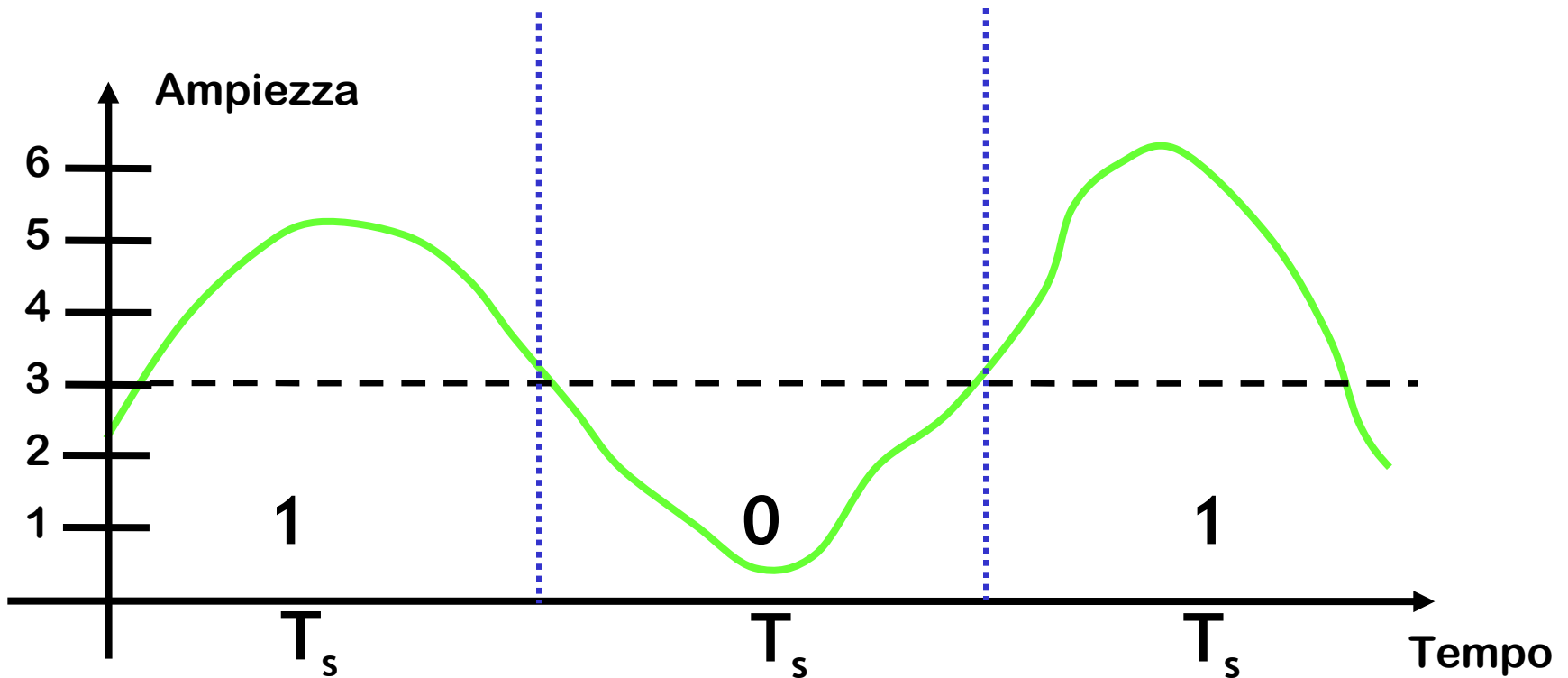
Segnale Digitale Ideale

- Segnale che può assumere solo due valori associati rispettivamente a “0” e “1”
 - Al valore “0” si associa, per esempio, 0 V
 - Al valore “1” si associa, per esempio, 5 V



Segnale Digitale Reale

- Fisicamente si riesce a generare qualcosa di simile...(un segnale analogico continuo che varia approssimativamente tra due valori)



Osservazione

- **Tutti i segnali, dunque, sono “analogici” a livello fisico, poichè non è possibile variare in maniera istantanea il valore di una grandezza fisica.**
- **Tuttavia in un segnale digitale non conta il valore del segnale in un dato istante (come nel segnale analogico *propriamente detto*), ma se questo è maggiore o minore di un certo valore soglia.**
- **Vi sono solo 2 “stati” significativi**

Segnali a più dimensioni

- **Immagine** = luminosità e colore sono **funzione delle coordinate spaziali**:

$$S(\mathbf{x}, y)$$





Video, Animazioni

- Video (ma anche immagini animate, etc...) = sequenza di immagini → c'è anche il tempo

$$S(x, y, t)$$



Fotogramma 0

$$t = t_0$$

...

....



Fotogramma 4000

$$t = t_0 + 4000 * \Delta T$$

Immagini digitali

Le **immagini digitali** sono composte da **matrici di pixel** (i **quadrati** visibili nell'ingrandimento mostrato nella figura a destra):

PIXEL = **P**icture **E**lement (**E**lemento immagine)

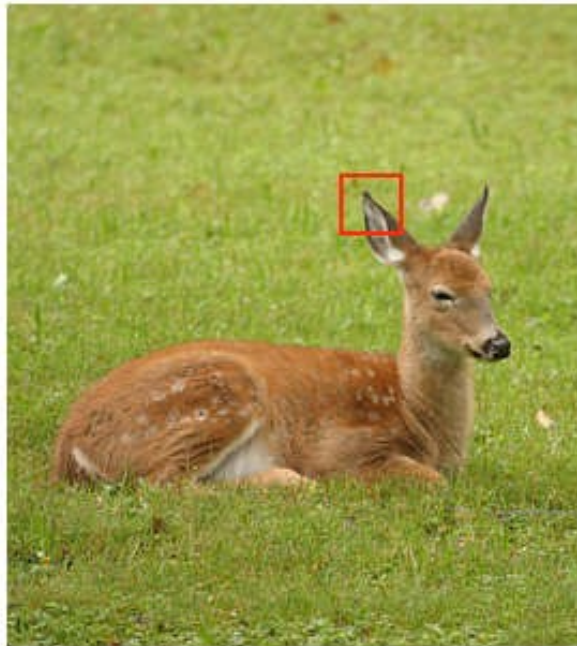
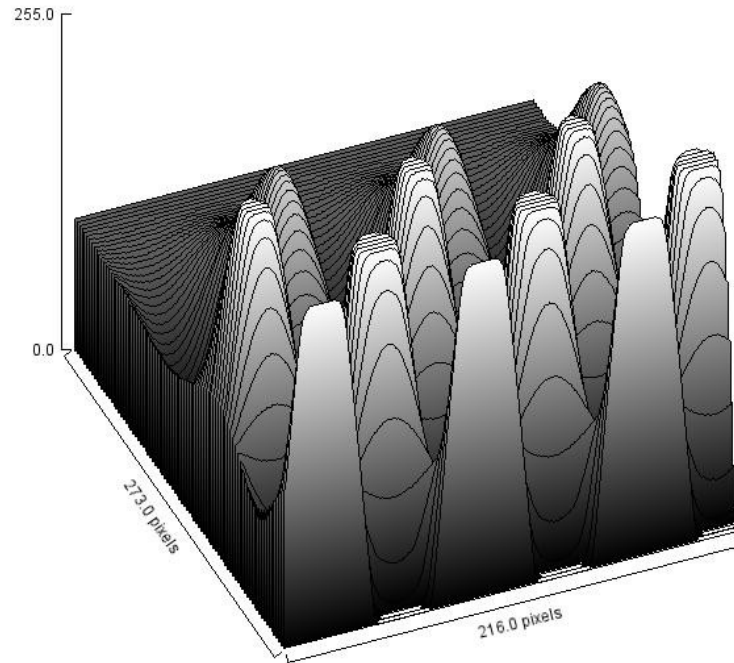
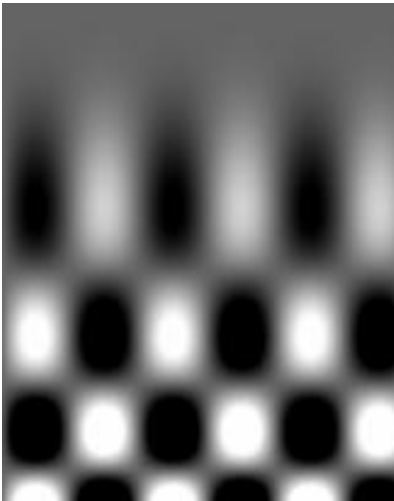


Immagine in scala di grigio = $L(x,y)$

Immagine come funzione bidimensionale

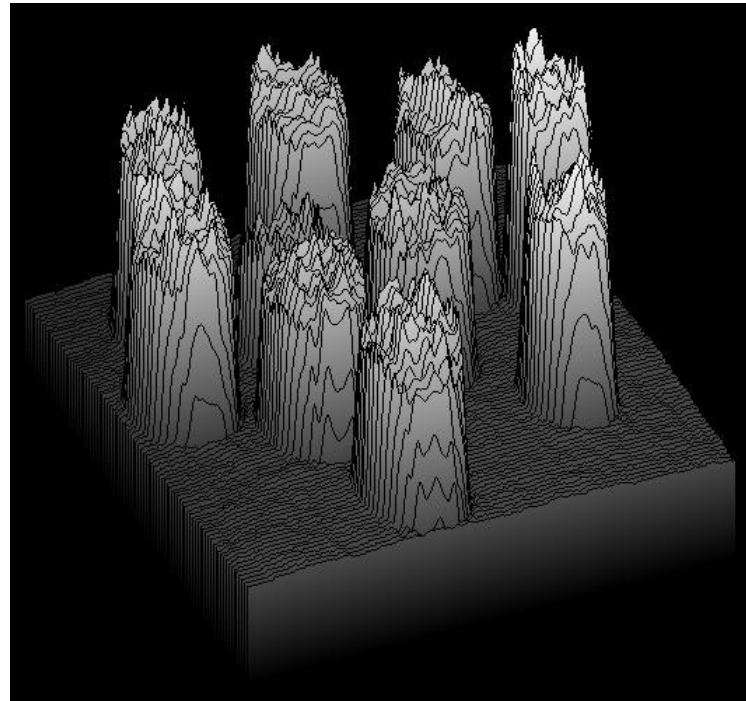


Nel caso di un'immagine in scala di grigio come quella mostrata a sinistra il contenuto dell'immagine è rappresentato dal valore di luminosità (opportunamente **quantizzato con 256 livelli**) per ogni pixel.

Tale luminosità è rappresentata sull'asse z del grafico a destra in funzione di x e y (che rappresentano le coordinate del pixel anche queste discrete)

Immagine in scala di grigio = $L(x,y)$

Immagine come funzione bidimensionale

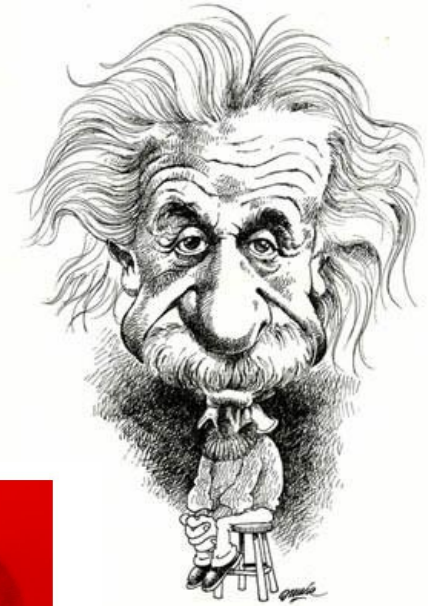


Immagini digitali

- Nel caso di un'immagine in bianco e nero ad ogni pixel è associato un solo valore che può essere 0 o 1
- Nel caso di un'immagine in gradazione di grigio per ogni pixel ho un valore (luminosità quantizzata) che in genere è un numero naturale compreso tra 0 e 255 (1 byte/pixel)
- Nel caso di un'immagine a colori per ogni pixel ho una terna di valori invece di uno soltanto compresi tra 0 e 255, in pratica è **come se avessi 3 segnali invece di uno** soltanto



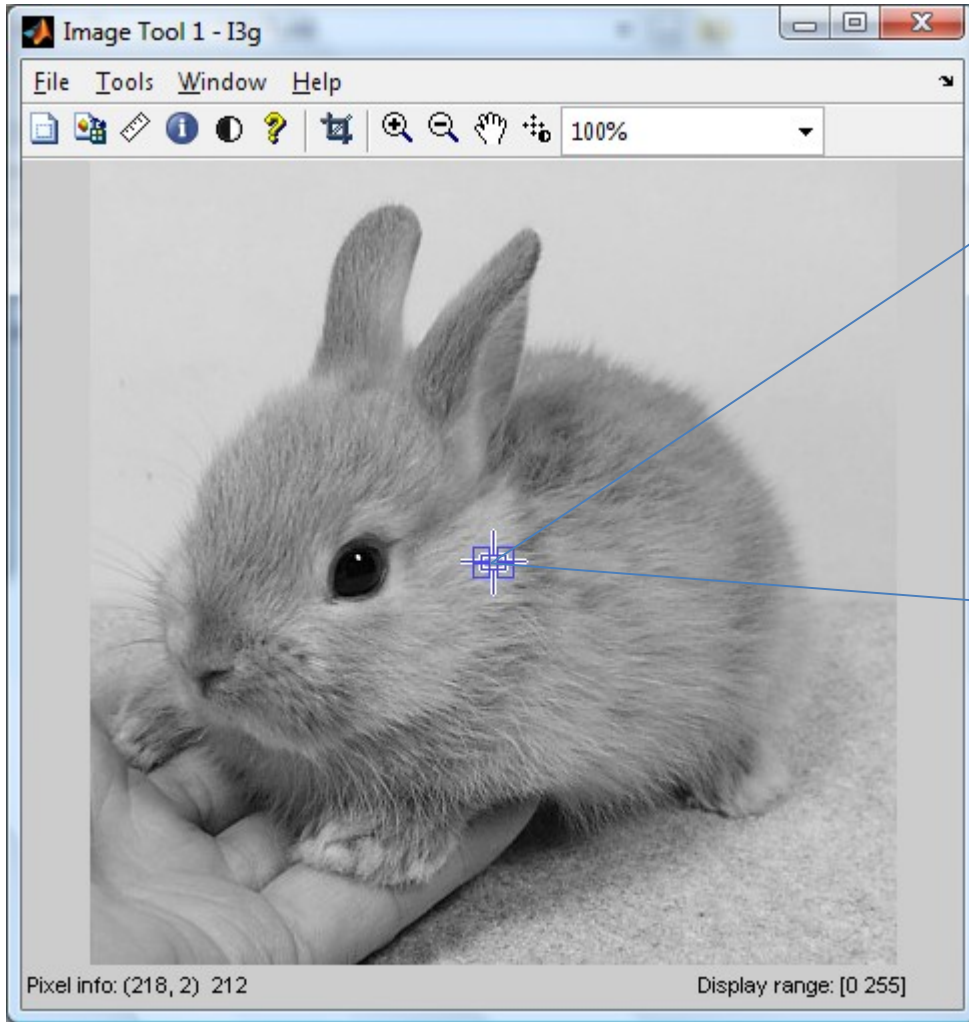
Bianco e
Nero:
 $L(x,y) = [0,1]$



Gradazioni di grigio : $L(x,y) = [0,255]$



A colori RGB :
 $R(x,y), G(x,y), B(x,y)$

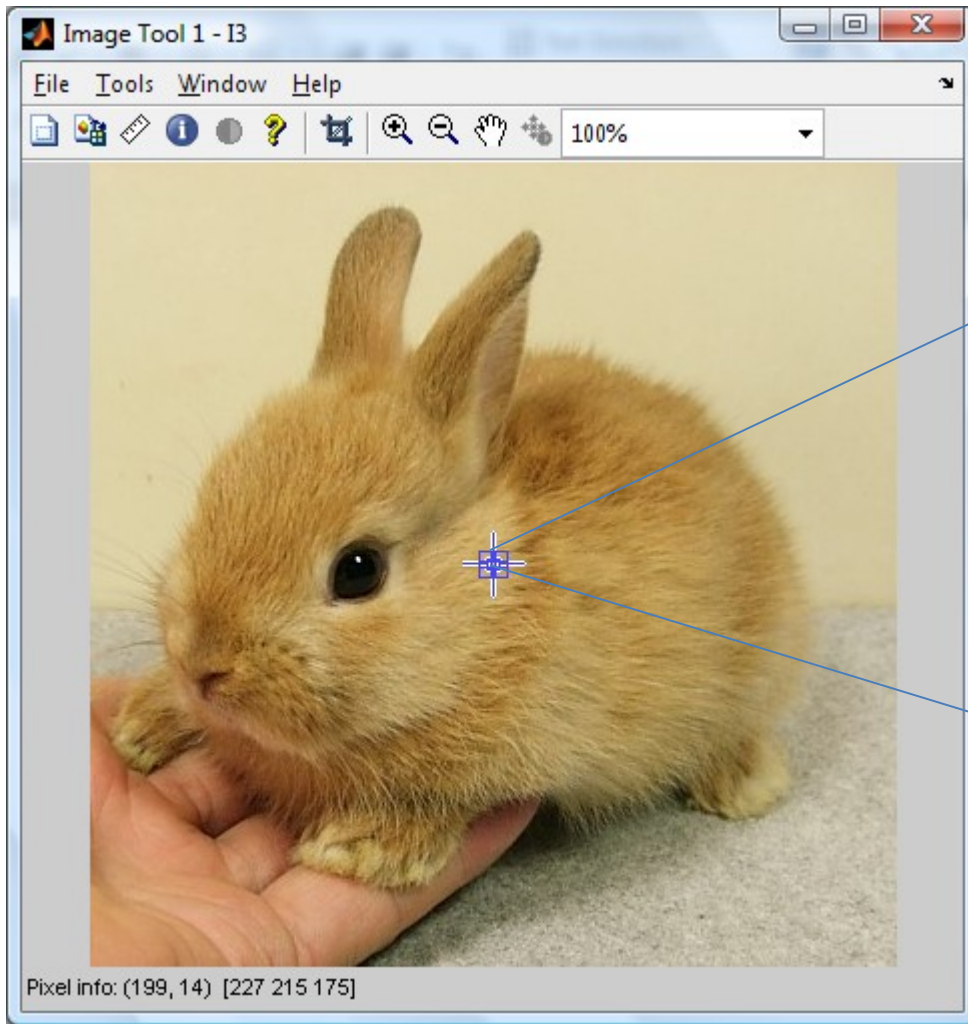


Pixel Region (Image Tool 1)

File Edit Window Help

173	178	178	177	174	171	164	157	151	149	154
174	176	173	169	173	173	165	159	160	159	161
168	169	166	161	170	178	178	180	187	177	165
156	167	173	173	170	176	180	180	178	177	176
167	172	172	168	164	169	168	162	162	169	174
165	168	165	161	158	163	161	154	156	168	173
166	169	168	166	162	165	165	162	165	170	170
166	167	165	162	169	167	166	168	169	166	163

Pixel info: (200, 203) 161



Pixel Region (Image Tool 1)

File Edit Window Help

197	R:192	R:199	R:207	R:207	R:209	R:207
164	G:159	G:169	G:177	G:177	G:179	G:177
95	B: 90	B: 99	B:107	B:105	B:107	B:107
204	R:204	R:201	R:207	R:211	R:211	R:211
171	G:171	G:168	G:174	G:178	G:178	G:178
100	B:100	B: 97	B:103	B:107	B:107	B:107
203	R:199	R:195	R:200	R:199	R:193	R:193
170	G:166	G:162	G:167	G:166	G:160	G:160
99	B: 95	B: 91	B: 96	B: 95	B: 89	B: 89
196	R:192	R:189	R:194	R:192	R:185	R:185
163	G:159	G:156	G:161	G:159	G:152	G:152
92	B: 88	B: 85	B: 90	B: 88	B: 81	B: 81

Pixel info: (199, 202) [203 170 99]

animazione RGB

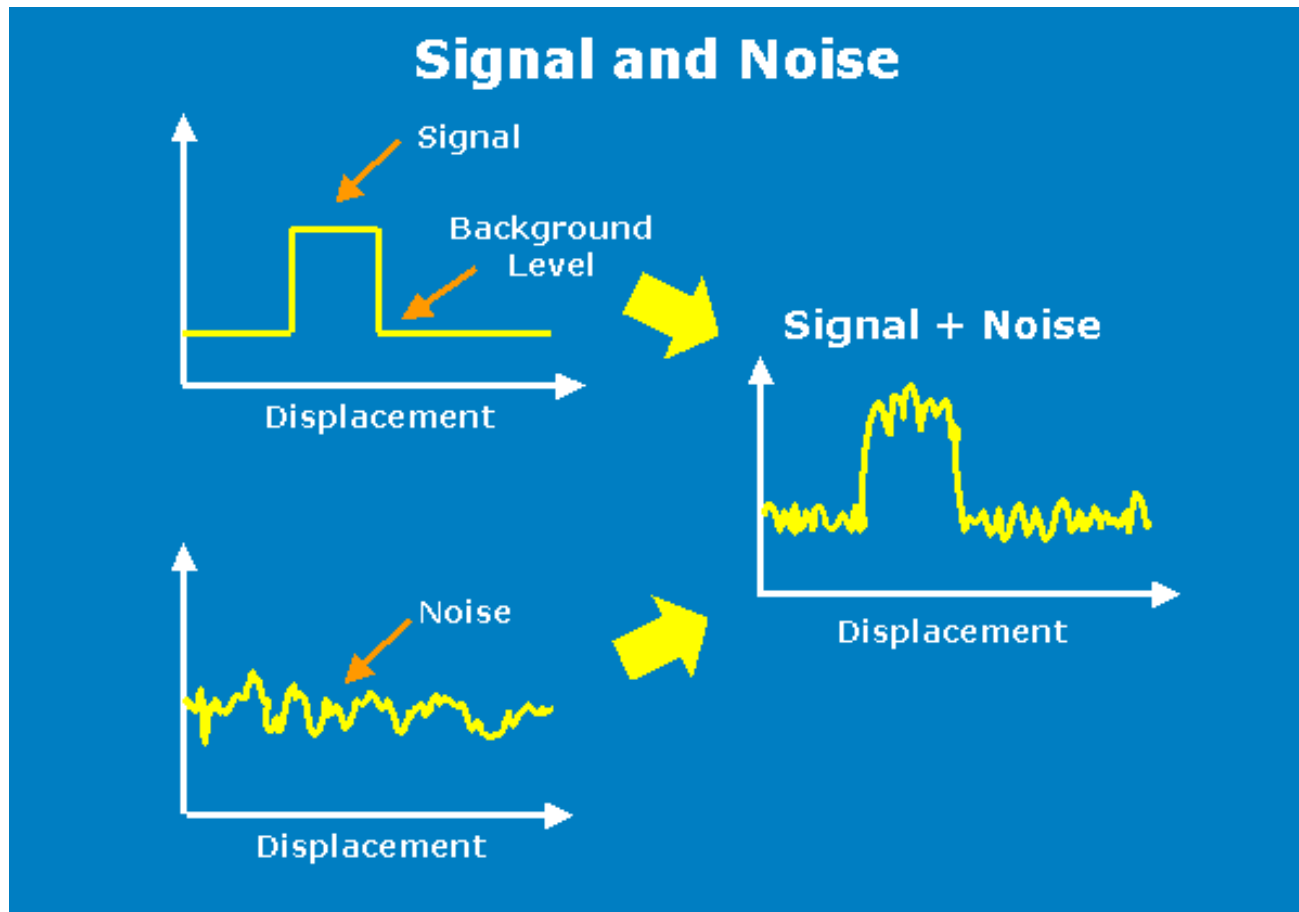
Rumore/Disturbo

- Qualsiasi **segnale indesiderato** che si sovrappone al segnale utile alterandolo
- **Fonti:**
 - **Interne** alle apparecchiature che generano, trasmettono e ricevono il segnale, in tal caso si parla più propriamente di **Rumore**
 - **Esterne:** interferenze elettromagnetiche, multipath fading ... in tal caso si parla di **Disturbo**

Esempi di “disturbi”

- Voce di altre persone che si sovrappone a quella dell'interlocutore
- Copie “ritardate” dello stesso segnale (frequente nelle comunicazioni wireless noto come [multipath fading](#))

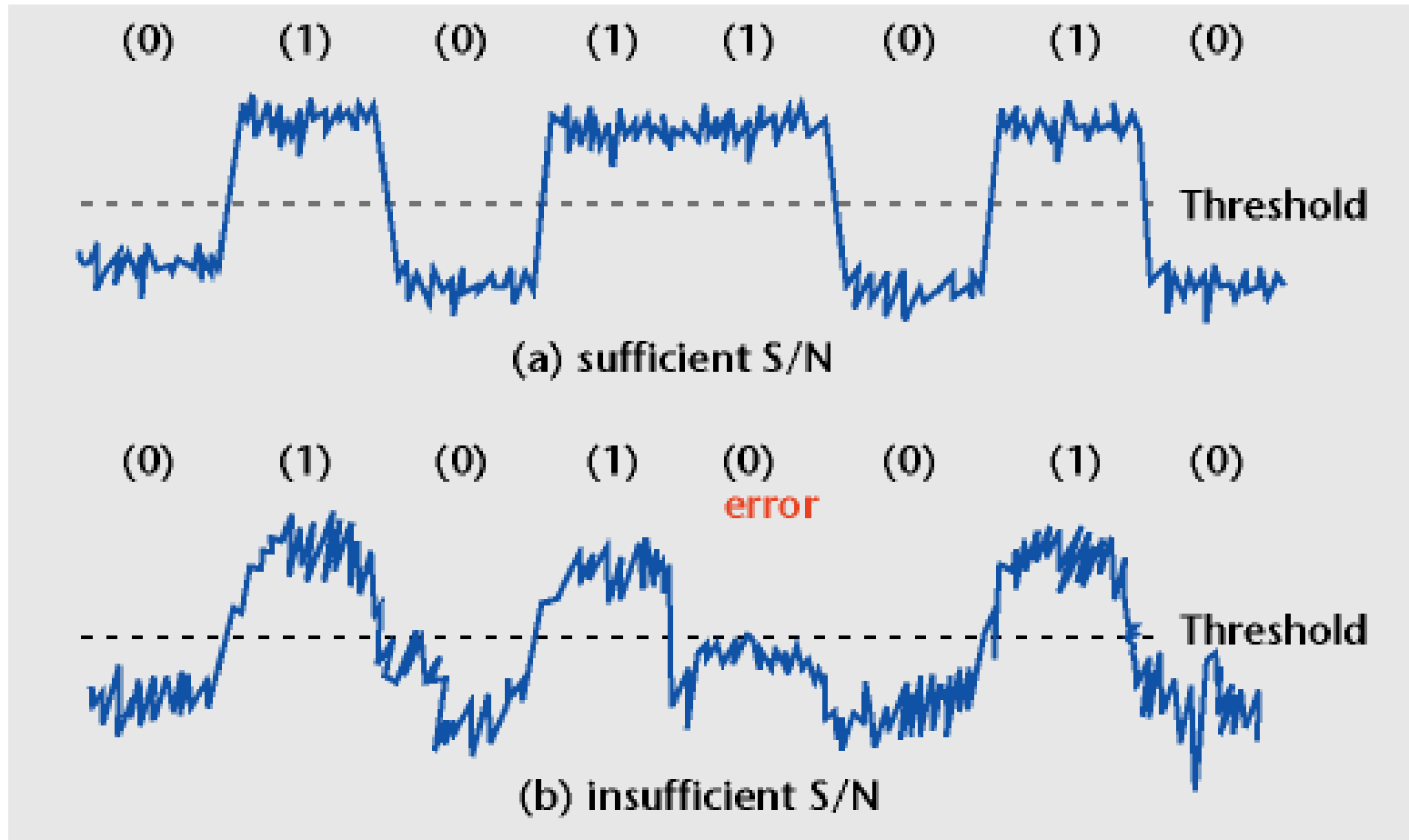
Rumore/Disturbo



Rumore/Disturbo

- Produce una variazione del segnale
- Se tale variazione è piccola rispetto all'ampiezza del segnale, l'effetto può essere trascurato,...
- in caso contrario il segnale può essere alterato al punto da essere irriconoscibile (e quindi inutilizzabile) dal ricevente.

Rapporto segnale/rumore

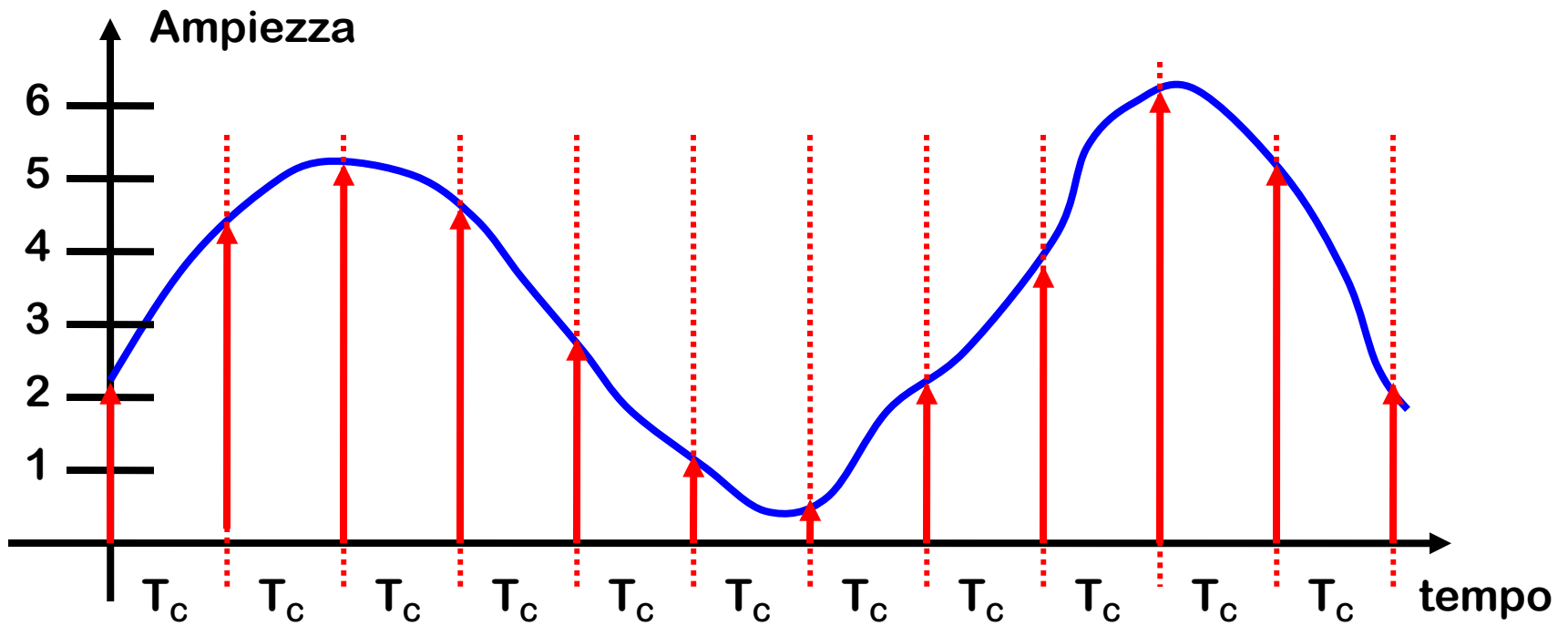


Conversione Analogico-Digitale

- La conversione da un segnale analogico in digitale può essere scomposta, *a livello di principio*, nelle seguenti operazioni:
 - **Campionamento**
 - **Quantizzazione**
 - **Conversione binaria**
 - **Generazione degli impulsi**

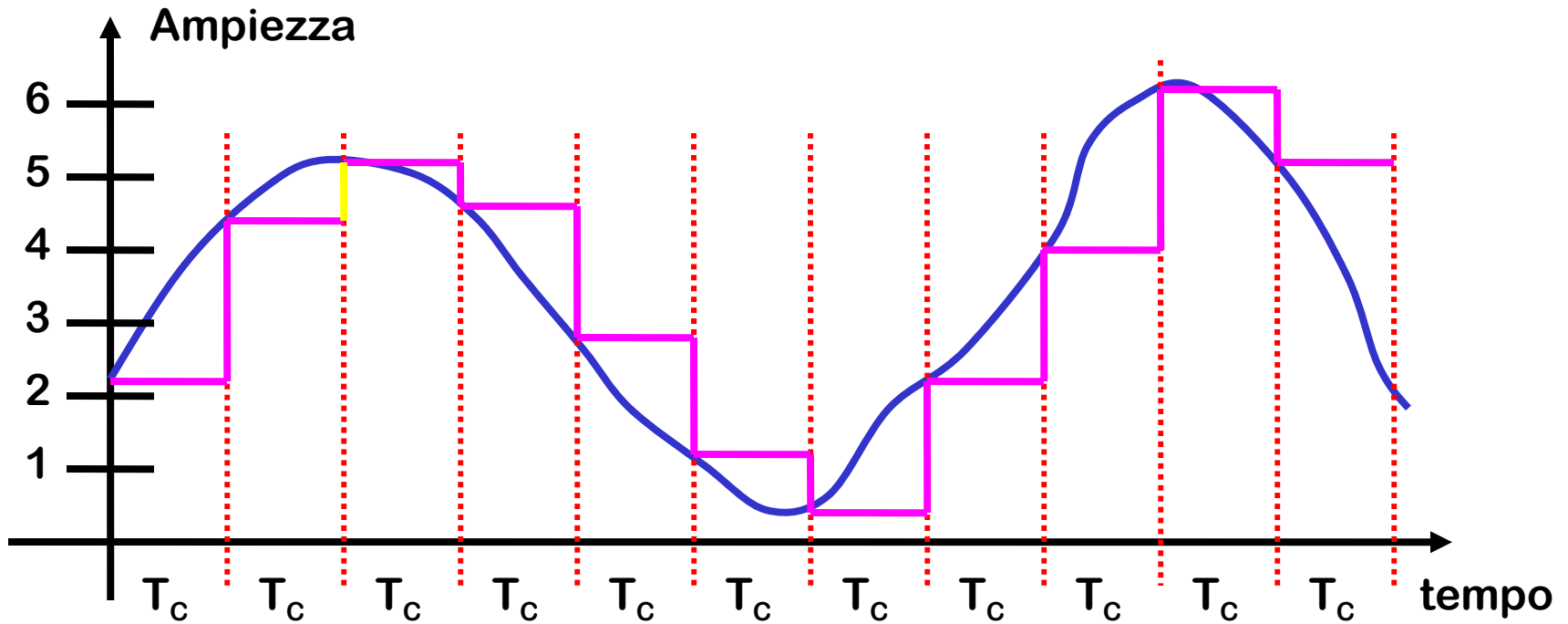
Campionamento

- Per effettuare la conversione analogico-digitale si “campiona” il segnale ad intervalli di tempo regolari T_c , ovvero si considera il valore che il segnale analogico assume nell’istante di campionamento



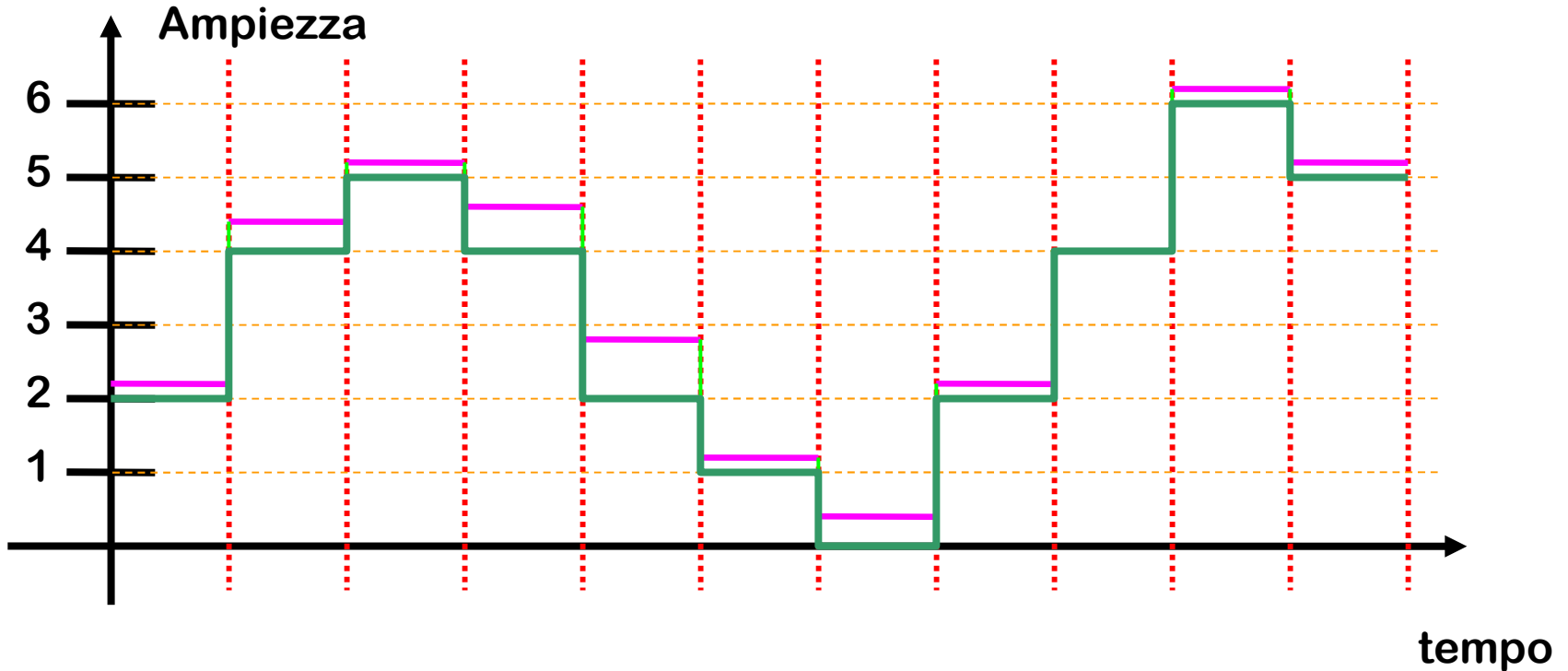
Campionamento

- Si approssima dunque il segnale originario con i valori assunti negli istanti di campionamento (*sampling*). E' bene che T_c sia sufficientemente piccolo in maniera da permettere di avere un'approssimazione adeguata del segnale originario



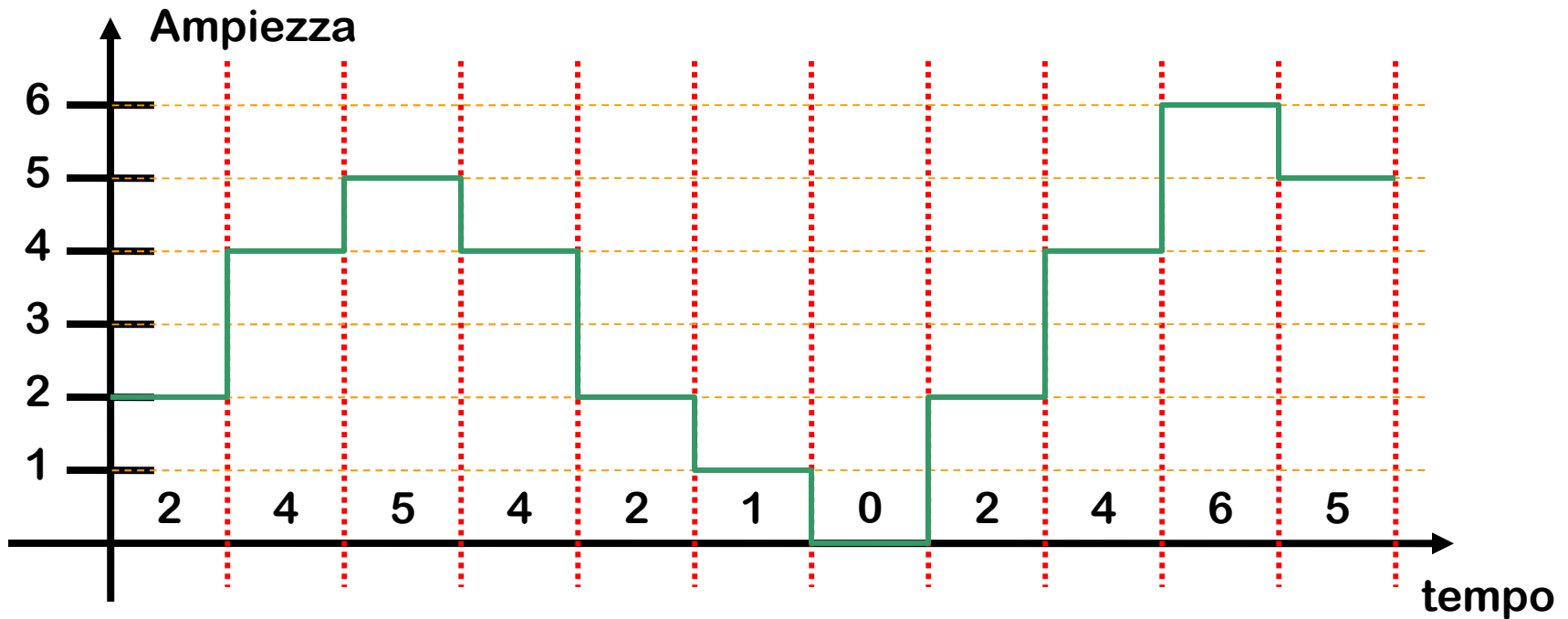
Quantizzazione

- Il valore ricavato viene ulteriormente approssimato in maniera da utilizzare solo un insieme limitato di valori (in questo caso 0,1,2,3,4,5,6).



Quantizzazione

- Il segnale ricavato attraverso i passi suddetti viene detto “*quantizzato*”. Esso può essere rappresentato con la sequenza di valori numerici che definiscono l’ampiezza del segnale nei vari intervalli. Es. 2,4,5,...



Codifica binaria

- A questo punto è necessario convertire i valori così ottenuti in binario e generare il segnale digitale
- Avendo 7 valori possibili 0,1,...6 ho bisogno di almeno 3 bit(8 combinazioni possibili: 000,001, ,111) in maniera che ad ogni valore possa essere associata una sequenza «unica» di bit:
- Es. 000 \rightarrow 3 001 \rightarrow 4, 010 \rightarrow 2, ...
- Notare che non è necessario eseguire una conversione binaria del valore numerico, semplicemente è sufficiente stilare una tabella in cui si decide quale valore corrisponde ad ogni sequenza binaria
- Delle otto sequenze di bit, utilizzerò solo le prime 7 ignorando l'ultima (111) avendo solo 7 valori e non 8!

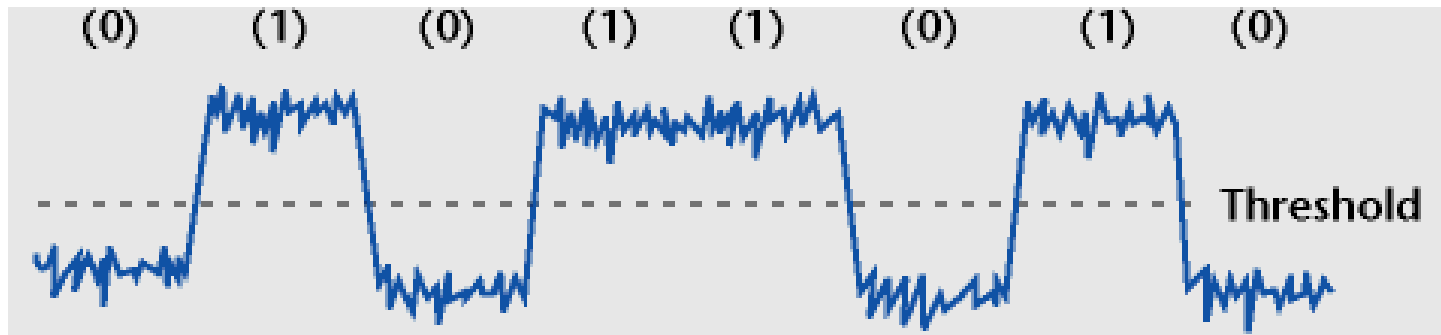
Codifica binaria

- Ogni valore «quantizzato» (ovvero appartenente all'insieme finito prescelto) sarà codificato in una sequenza binaria(3 bit nell'esempio precedente) e tutte le sequenze generate in successione per ogni campione insieme formeranno il messaggio trasmesso al destinatario
- Il messaggio assumerà dunque la forma di una lunga sequenza binaria del tipo:

010100101100010001000...

Generazione degli impulsi

- A questo punto la sequenza di prima viene trasformata in una serie di impulsi simile alla seguente



Animazioni

- Conversione analogico digitale (variare numero bits e frequenza di campionamento, cliccare su samples per visualizzare i campioni)

Esempio

- Quando si legge che un file **MP3** è a **128 bit** significa che si utilizzano 128 bit per rappresentare l'ampiezza del segnale, ovvero i valori possibili sono 2 elevato a 128.
- Il **range** potrebbe essere $[0,1]$ dove 0 indica assenza di segnale e 1 la massima intensità raggiunta.
- Il segnale MP3 a **128 bit** è sicuramente migliore di uno a **96 bit** perché riesco ad approssimare meglio il segnale originario. **Il numero di bit è dunque un indice della qualità di riproduzione.**
- Lo svantaggio è che **aumentando il numero di bit** impiegato per rappresentare ogni singolo campione, **aumenta la dimensione del file.**

Conversione Digitale-Analogica

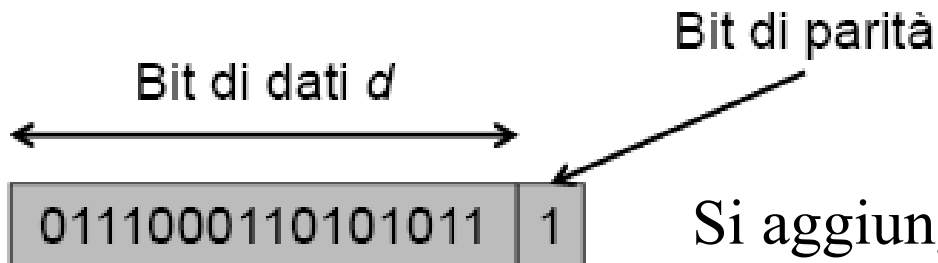
- Quando ascoltiamo un MP3 con il nostro PC o un brano musicale con un lettore CD viene effettuata l'operazione inversa.
- Un segnale digitale è convertito prima in un segnale analogico di natura elettrica (variazione di tensione) e infine, grazie alle casse audio, in una variazione di pressione ovvero in suono.

Rilevamento e correzione errori

- **Utilizzando più bit di quelli strettamente indispensabili è possibile aumentare la tolleranza ai disturbi,** servendosi di algoritmi in grado di rilevare e correggere gli errori eventualmente presenti nelle sequenze di bit ricevute
- **Poiché il segnale nel corso della trasmissione si attenua, l'incidenza dei disturbi aumenta** man mano che il segnale si propaga nel mezzo di trasmissione
- **Se il cammino è molto lungo, è necessario rigenerare il segnale** grazie a dispositivi che rilevano e correggono gli errori di trasmissione, ritrasmettendo amplificato e senza errori il segnale ricevuto

Rilevamento errori(eseempio): Bit Parità

- Supponiamo di dover trasmettere d bit
- Si trasmettono $d+1$ bits aggiungendo un bit determinato in maniera tale che i $d+1$ bit abbiano un numero pari di 1 come mostrato nella figura seguente



Si aggiunge un 1 alla fine
Numero di 1 = $9 + 1 = 10$ Pari

Bit Parità(esempi)

Bit Parità



000000100010	0	aggiungo 0 perché il numero di 1 è già pari Numero 1 = 2+0 = 2
100000100010	1	aggiungo 1 perché il numero di 1 è dispari Numero 1 = 3+1 = 4
100000110010	0	aggiungo 0 perché il numero di 1 è già pari Numero 1 = 4+0 = 4

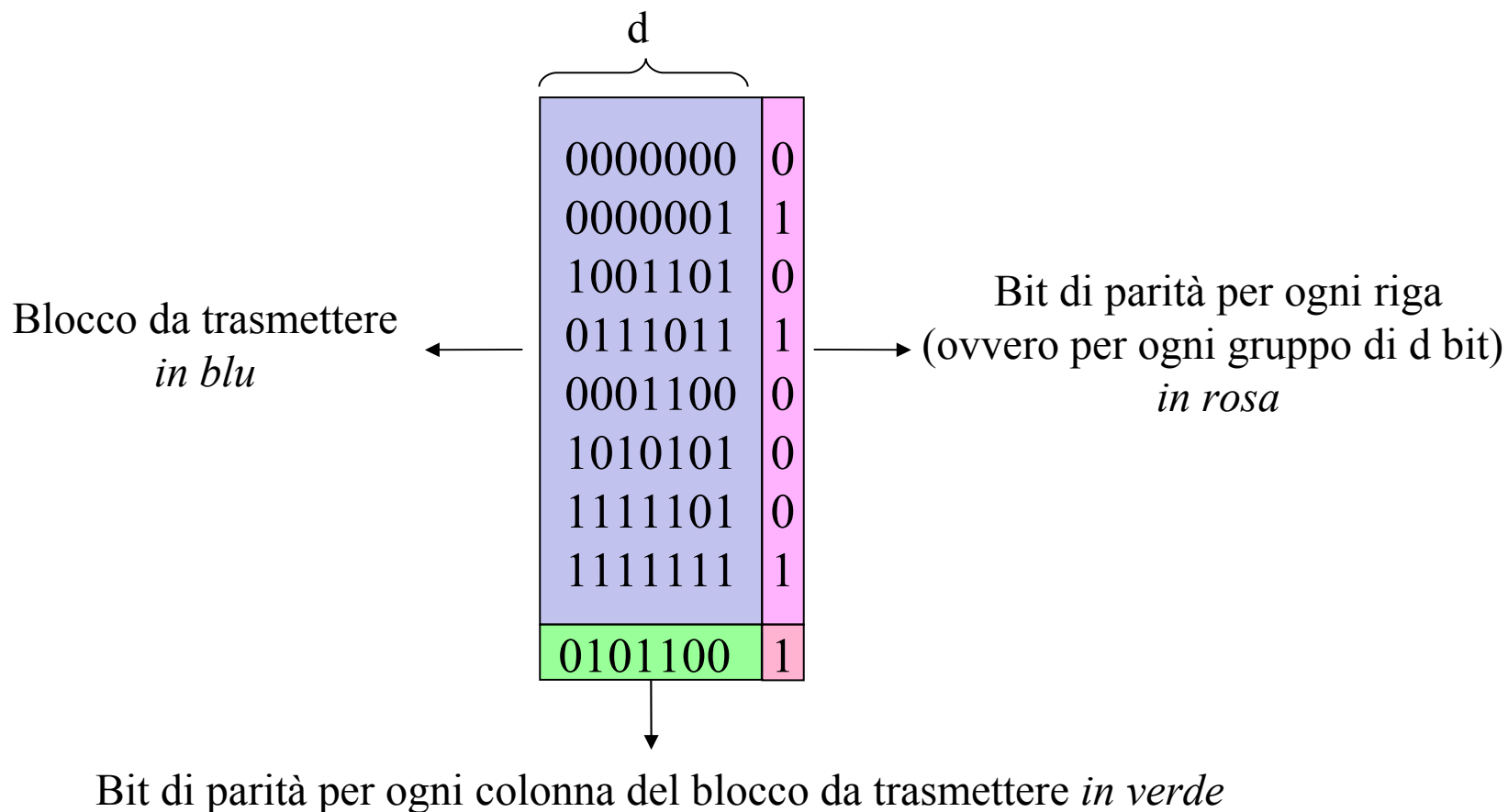
Bit Parità

- In questa maniera se durante la trasmissione un bit viene alterato passando da 0 a 1, o viceversa da 1 a 0, **siamo in grado di rilevare l'errore** perché il numero di 1 adesso risulterà dispari
- Tuttavia ***non*** possiamo sapere ***quale dei d bits*** trasmessi ***è stato alterato!!***
- **Per essere in grado di risalire alla sequenza di bits corretta, devo aggiungere ulteriori bits ai dati trasmessi**, questi bits come per il bit di parità non servono a trasmettere informazioni, ma semplicemente a permettere l'individuazione e la correzione degli errori di trasmissione

Rilevamento e correzione: Bit Parità a “blocchi”

- Uno dei metodi più semplici è un'estensione “a blocchi” del bit di parità
- Questo metodo funziona pensando di raggruppare n sequenze di d bits in un unico blocco
- **Per ogni “blocco” da trasmettere calcolo il bit di parità per ogni colonna e per ogni riga e trasmetto anche questi in sequenze di $d+1$ bits oppure separatamente**
- **In ricezione individuando in quale riga e in quale colonna i bit di parità (di riga e di colonna) non rendono pari gli 1 di ogni riga e di colonna è possibile individuare esattamente il bit alterato e correggere l'errore**

Rilevamento e correzione: Bit Parità a “blocchi”



Bit Parità a “blocchi”

Bits ricevuti (con errore) vengono mostrati disposti in blocco ma in realtà sono ricevuti in sequenza, riga per riga



Individuazione bit alterato

Correzione degli Errori

- **Analogamente** nel caso in cui alcuni bit siano alterati durante **il processo di scrittura su un supporto di memoria**, utilizzando “codifiche” più complesse della semplice conversione del valore in binario, posso risalire al valore originario e correggere l’errore