**RETI DI CALCOLATORI**

Sommario

[**LEZIONE 1 – INTRODUZIONE ALLE RETI** 2](#_Toc148795765)

[Tipologie di Reti 2](#_Toc148795766)

[Tecnologie di comunicazione 3](#_Toc148795767)

[Topologie 5](#_Toc148795768)

[Struttura di una rete 6](#_Toc148795769)

[**LEZIONE 2 – MODELLI DI RETE** 7](#_Toc148795770)

[Protocolli di comunicazione 7](#_Toc148795771)

[Il modello ISO/OSI 9](#_Toc148795772)

[Tipologie di Connessione 12](#_Toc148795773)

[Affidabilità di un servizio 13](#_Toc148795774)

[**LEZIONE 3 – MODELLO TCP/IP E CONFRONTO** 14](#_Toc148795775)

[Modello TCP/IP 14](#_Toc148795776)

[Livello Accesso Rete 15](#_Toc148795777)

[Livello Internet 15](#_Toc148795778)

[Livello Trasporto 15](#_Toc148795779)

[Livello Applicazione 15](#_Toc148795780)

[**LEZIONE 4 – LIVELLO FISICO E TEORIA DEI SEGNALI** 16](#_Toc148795781)

[Basi della trasmissione 16](#_Toc148795782)

[Mezzi di Trasmissione 17](#_Toc148795783)

[Tipologia di segnale 17](#_Toc148795784)

[Analisi Spettrale 17](#_Toc148795785)

[Teorema di Fourier 18](#_Toc148795786)

[Teorema di Nyquist 19](#_Toc148795787)

[Rapporto Segnale/Rumore 20](#_Toc148795788)

[Teorema di Shannon 20](#_Toc148795789)

[Trasmissione 21](#_Toc148795790)

[Mezzi trasmissivi 21](#_Toc148795791)

[Topologie di reti in fibra ottica 23](#_Toc148795792)

[Trasmissioni wireless 24](#_Toc148795793)

[Classificazione onde elettromagnetiche 24](#_Toc148795794)

[Trasmissione e modulazione 26](#_Toc148795795)

[**LEZIONE 5 - SISTEMA TELEFONICO E MODULAZIONE** 26](#_Toc148795796)

[**LEZIONE 6 – IL LIVELLO DATA LINK** 31](#_Toc148795797)

[Introduzione 31](#_Toc148795798)

[Progettazione del livello Data Link 31](#_Toc148795799)

[Operazioni del livello due 32](#_Toc148795800)

[Suddivisione in frame 33](#_Toc148795801)

[Rilevazione e correzione degli errori 34](#_Toc148795802)

[Codici a correzione di errore 34](#_Toc148795803)

[Codici a rilevazione d’errore 39](#_Toc148795804)

# **LEZIONE 1 – INTRODUZIONE ALLE RETI**

Il primo argomento da introdurre quando si parla di reti di calcolatori è la differenza tra essi e i sistemi distribuiti.

**Un sistema distribuito** è un insieme di computer indipendenti che appare ai propri utenti come un singolo sistema coerente. Un classico esempio di sistema distribuito è il World Wide Web.

Altri esempi:

GRID Computing: risorse di calcolo distribuite su diverse aree geografiche

▶ Cloud Computing: esecuzione remota di codice ed applicativi

▶ Edge Computing: esecuzione di operazioni remote nel punto di

acquisizione dei dati

**In una rete di calcolatori** mancano la coerenza, il modello ed il software tipico di un sistema distribuito. Gli utenti della rete vedono i singoli componenti di essi, un utente difatti per eseguire un programma che si trova su una specifica remota della rete deve collegarsi manualmente ad essa per eseguirlo.

## Tipologie di Reti

Suddividiamo principalmente due tipologie di reti:

**Cablate:** (Rame, fibra ottica) da cui discendono le seguenti reti:

* LAN (Local Area Network) 🡪 rete locale/aziendale che va dai 10m – 1km
* MAN (Metropolitan Area Network) 🡪 rete metropolitana che va dai 100m – 10km
* WAN (Wide Area Network) 🡪 rete geografica che va dai 10km – 1000km

**Wireless**: (Radiofrequenze, infrarossi) da cui discendono le seguenti reti:

* WPAN (Wireless Personal Area Network) 🡪 rete di piccolissime distanze (cm). Rete tipica di NFC, Bluethoot, IR
* WLAN (Wireless Local Area Network) 🡪 come rete LAN, ma senza fili.
* WMAN (Wireless Metropolitan Area Network) 🡪 come rete MAN, ma senza fili.
* WWAN (Wirless WAN) 🡪 rete geografica wirless

Diverse tipologie di reti possono essere interconesse tra di loro formando reti sempre più grandi (Internet)

Immagine che contiene schermata, diagramma

Descrizione generata automaticamente

## Tecnologie di comunicazione

Le reti di calcolatori dispongono di un’altra caratteristica principale: la tecnologia di trasmissione.

Le tecnologie di trasmissione possiamo dividerle in due tipi:

**1.Trasmissione punto a punto**: Ogni calcolatore deve connettersi direttamente ad un altro calcolatore. Se due calcolatori non sono direttamente collegati, è necessario creare un instradamento attraverso altri calcolatori. Bisogna sempre raggiungere un nodo nella rete.

Risulta essere costoso per via delle numerose connessioni dedicate.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page8image29424048page8image13301760page8image13301200 | | | |
|  |  | |  |

**2.Trasmissione broadcast**: sono dotate di un unico canale di comunicazione (bus). Ogni calcolatore ha un identificativo univoco (indirizzo di rete)

Il pacchetto viene inviato da una macchina a tutte le altre macchine, ma viene letto solamente dal destinatario esaminando il proprio indirizzo di rete.

Alcuni sistemi broadcast supportano la trasmissione a un sottoinsieme delle macchine, tale tecnologia viene chiamata multicast. Infine, la connessione di due o più reti è chiamata internetwork.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page9image29416560page9image13332064page9image13332960 | | | |
|  |  | |  |

Non tutti i calcolatori possono trasmettere simultaneamente sul bus e quindi è necessario un sistema di regole che si dividono in:

**Regole statiche:** vengono prefissate e non possono cambiare nel tempo come ad esempio una politica di round robin con time slicing.

Ci saranno, ovviamente, spreco di tempo e risorse se un calcolatore non deve trasmettere.

**Regole dinamiche:** Di volta in volta si decide chi può utilizzare il mezzo di trasmissione.

Possono essere centralizzate (un’ unità centrale decide chi può iniziare la trasmissione) o distribuite (un calcolatore può decidere se trasmettere in base allo stato del mezzo).

Possibili collisioni.

## Topologie

La topologia è un concetto diverso dalla tipologia.

La topologia definisce come gli apparati di rete sono collegati tra di loro. Ogni elemento connesso nella rete è detto nodo e l’’informazione scambiata è detta pacchetto.

Suddividiamo topologia logica (come i dati vengono scambiati tra i nodi) e topologia fisica (dislocazione fisica dei noti. Ogni topologia risulterà essere un grafo)

Esistono varie topologie fisiche tra cui:

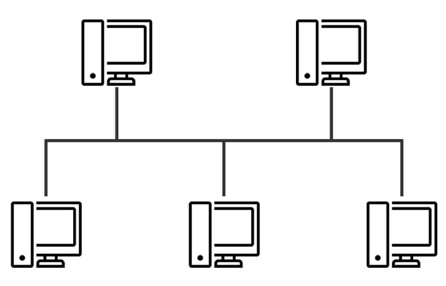
**Rete a bus**:

Le informazioni viaggiano su un unico canale.

Tutti i nodi possono leggere le informazioni in viaggio. In particolare, un nodo NON destinatario riceve il pacchetto, ma lo scarta. Un nodo destinatario lo legge.

PRO: Semplice da realizzare e da estendere.

CONTRO: Velocità ridotte.



**Rete ad anello:**

Le informazioni viaggiano su un unico canale.

Ci sono due modalità:

**Unidirezionale:** i pacchetti sono trasmessi in senso orario o antiorario.

**Bidirezionale:** i pacchetti sono trasmessi in entrambe le direzioni

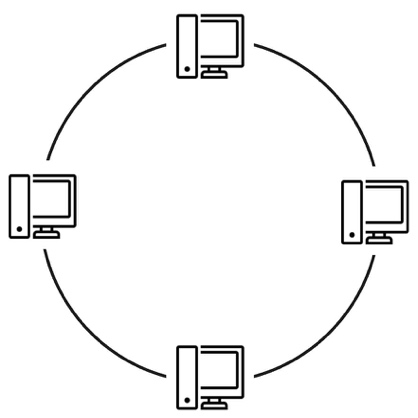
Nella rete ad anello, un nodo NON destinatario inoltra al successivo il pacchetto.

Un nodo destinatario lo legge e blocca l’inoltro.

Se il pacchetto torna al mittente, la comunicazione si interrompe ( destinatario non trovato)

PRO: Semplice da estendere e veloce

CONTRO: Bassa tolleranza ai guasti.



**Rete a stella:**

Esiste un nodo centrale che gestisce la comunicazione.

Le informazioni vengono inviate ad un nodo centrale che funge da “router” e indirizza il pacchetto verso il destinatario.

PRO: Semplice da realizzare, buone velocità

CONTRO: Tolleranza ai guasti parziale.

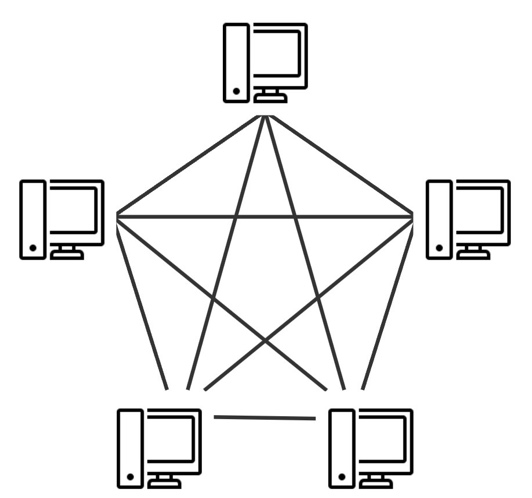
|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page14image29645936page14image16050912page14image16054384 | | | |
|  |  | |  |

**Rete a maglia (mesh)**

Si tratta di una rete in cui tutti i nodi sono collegati tra loro e ciascuno di essi ne riesce a raggiungere un altro attraverso un solo passaggio. Nel caso in cui uno dei cavi dovesse rompersi sarebbe possibile comunque l’arrivo a destinazione dei pacchetti. Si viene dunque a formare una maglia con percorsi multipli tra i nodi.

Tutti i nodi possono leggere le informazioni in viaggio.

Un nodo non diestinatario riceve il pacchetto, ma lo scarta, mentre il nodo destinatario lo legge.



PRO: Massima velocità e tolleranza ai guasti.

CONTRO: Costo elevato e difficile realizzazione.

## Struttura di una rete

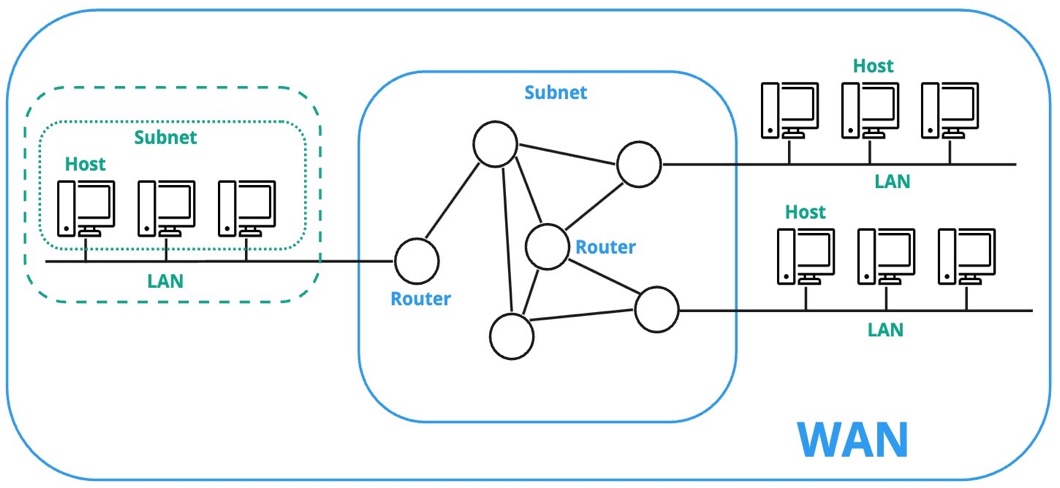
Una rete è strutturata dai seguenti componenti:

**Host** (unità di calcolo)

**Subnet** (Sottorete composta da tutti i nodi di una singola rete)

**Mezzo di trasmissione**

**Dispositivi di rete** che si occupano di instradare la comunicazione nella sottorete o verso le reti esterne 🡪 Modem, Gateway, Router, Switch ecce cc.



* Anche se esistono regole standard per le reti queste sono eterogenee tra di loro
* Una rete può comunicare con un’altra di diverso tipo
* Possono inoltre comunicare reti basate su mezzi di trasmissione differente

▶ Cablata <—> Satellite

▶ Radio <—> Cablata

▶ Satellite <—> Radio

• Internetwork - collega reti di diversa tipologia utilizzando dei router multiprocollo detti gateway

* + ▶  LAN <—> MAN <—> WAN
  + ▶  I gateway effettuano sia il routing (instradamento) sia la conversione

di protocollo.

# **LEZIONE 2 – MODELLI DI RETE**

## Protocolli di comunicazione

Visto che la rete è eterogenea (diverse tipologie e tecniche di comunicazione e scambio dati) è necessario definire un insieme di regole specifiche. Tali regole sono dette protocolli.

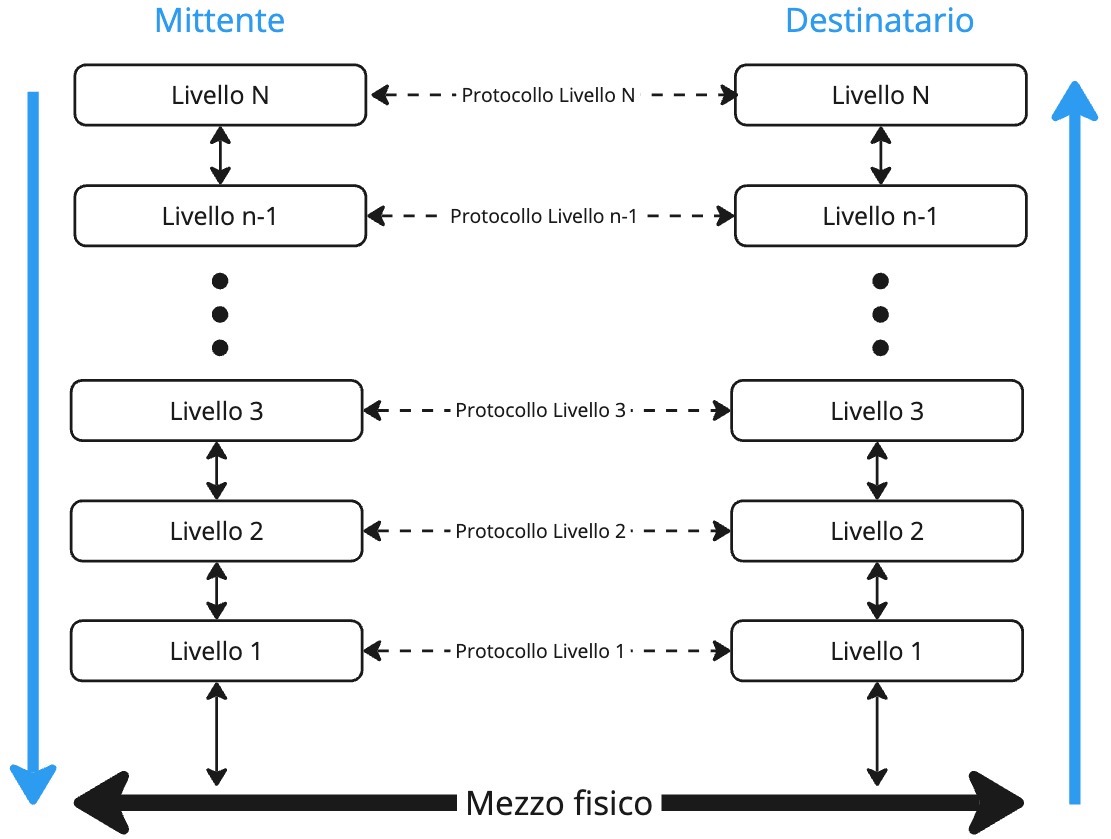
**Def: Insieme coordinato di regole che consente a due interlocutori (un utente e un calcolatore elettronico, due utenti oppure due calcolatori) di scambiarsi rapidamente e univocamente dati e messaggi.**

Una rete principalmente è organizzata a livelli (o strati) dove ogni livello ha una propria responsabilità nella rete.

Ogni elemento di un livello **n** comunica con lo strato successivo **n+1 (service user)** e quello precedente **n-1** **(service provider)** mediante interfacce di comunicazione dette **SAP** (un livello NON può comunicare con livelli non adiacenti)

Ogni livello interagisce con il suo corrispettivo sull’host mittente/destinatario.

L’insieme dei protocolli di una singola architettura è detto **Stack di protocolli**.



COME AVVIENE LA COMUNICAZIONE TRA LIVELLI ?

Facciamo un esempio con un’architettura a 5 livelli (N=5)

1) Un messaggio M parte dal livello più alto (n=5).

2) Il messaggio viene inviato al livello n-1 (4)

3) Il livello 4 inserisce un header con informazioni di controllo (H4).

4)Si passa il pacchetto al livello n-1 (3)

* Esiste una dimensione massima per ogni pacchetto
* Il pacchetto deve essere scomposto in pacchetti più piccoli (M1, M2).
* Nuove informazioni di controllo vengono aggiunte ad ogni porzione (H3) N.B. Le informazioni di controllo precedenti non vengono suddivise!

5) Il pacchetto viene inviato al livello n-1 (2). Vengono aggiunte informazioni di controllo (H2) ed informazioni dette trailer per determinare il termine del pacchetto (T2).

6) Il messaggio può essere inviato tramite mezzo fisico.

7) Si risale l’albero verso il livello più grande

8) Ad ogni livello n+1 vengono lette le informazioni di controllo.

N.B. Le informazioni di controllo sono utilizzate dai protocolli dello stesso livello

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page7image29682656page7image15803360page7image15805824 | | | |
|  |  | |  |

Ogni livello ha un determinato compito, es:

* **Indirizzamento**,
* **Controllo errori di trasmissione**
* **Frammentazione**
* **Controllo degli errori**
* **Trasferimento di dati (Simplex, Half-Duplex, Full-Duplex)**
* **Multiplexing**

Le operazioni che un livello effettua per utilizzare un servizio sono dette **primitive** e coinvolgono il livello n e i suoi adiacenti.

Ogni primitiva ha associata una corrispettiva risposta con il livello da raggiungere.

## Il modello ISO/OSI

Il **modello ISO/OSI** è il modello di riferimento per le architetture di rete, difatti è chiamato standard de iure (standard di diritto poiché è nato dopo lo standard TCP/IP 🡪 de facto) e si tratta solo di uno schema concettuale che i produttori di hw di rete devono seguire.

E’ un modello composto da 7 livelli (3 fisici, 4 applicativi).

Quando il mittente spedisce verso il destinatario si tratta di incapsulamento poiché verranno aggiunte informazioni.

Quando il destinatario riceve il messaggio si tratta di decapsulamento poiché ricostruisce il messaggio.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page13image29732432page13image15755328page13image15745360 | | | |
|  |  | |  |

SINTESI DEL FUNZIONAMENTO DEI LIVELLI:

|  |  |  |
| --- | --- | --- |
| # | Livello | Definizione |
| 7 | Applicazione | Interfaccia tra il sistema di comunicazione e le applicazioni |
| 6 | Presentazione | Formatta e trasforma i dati in base la loro rappresentazione locale. Fornisce anche la cifratura/decifratura dei dati |
| 5 | Session | Si occupa delle sessioni di comunicazione, dall’inizializzazione alla chiusura |
| 4 | Trasporto | Invio e ricezione dei dati. Controllo e correzione (se possibile) degli errori |
| 3 | Rete | Creazione dei pacchetti, indirizzamento ed instradamento degli stessi ad alto livello (astrazione) |
| 2 | Data Link | Definizione del frame e dell’indirizzamento in funzione del mezzo fisico |
| 1 | Fisico | Trasmissione dei dati tramite il mezzo fisico |

**Livello Fisico:** Il livello fisico si occupa della trasmissione dei dati grezzi (bit) su un canale di comunicazione.

Specifica le caratteristiche meccaniche, elettriche e procedurali dell’apparato di connessione.

Inoltre, specifica le caratteristiche del mezzo fisico tra cui le tensioni scelte, durata di un singolo bit, tipo di trasmissione ecc.

Problemi tipici di questo livello sono i tempi di trasmissione del segnale, come avviene l’inizio e la fine della comunicazione o se la comunicazione può avvenire simultaneamente in entrambe le direzioni.

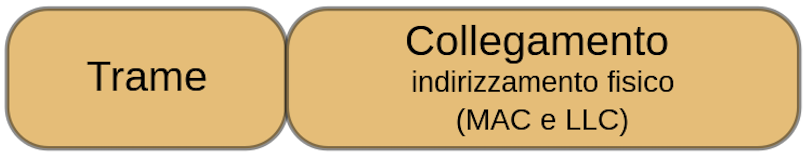
BIT 🡪 FISICO(Mezzo, Segnale e trasmissione binaria)

**Livello Data Link:** Il livello data link si occupa di trasformare i pacchetti ricevuti dal livello Network in frame che vengono inviati in sequenza al livello fisico.

Al termine del framing i dati diventano parte di un **nuovo pacchetto, dotato di un’intestazione (header) e di una coda (tail)**, che hanno la funzione anche di sequenza di controllo. Per ogni pacchetto ricevuto **il destinatario trasmette al mittente un segnale di ACK** (*Acknowledgment*), ovvero di conferma di ricevuta.

In questo modo il mittente è in grado di **capire quali pacchetti siano o meno arrivati a destinazione**. Nel caso di pacchetti corrotti, incompleti, persi o mal trasmessi, il mittente deve occuparsi della loro ritrasmissione.

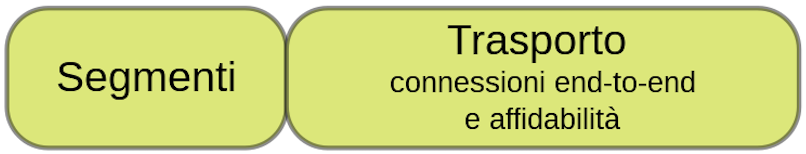
Il livello 2 **si occupa anche del controllo di flusso dei dati** e di tutta una serie di interventi correttivi nel caso in cui venga registrato uno sbilanciamento della velocità di trasmissione mediante un sottolivello chiamato MAC (Media Access Control).



**Livello di Rete (Network Layer):** Questo livello si occupa del routing dei dati attraverso una rete composta da più collegamenti e nodi. I principali compiti includono l'instradamento dei pacchetti, la gestione degli indirizzi IP e il controllo del traffico per evitare congestioni. Il protocollo IP (Internet Protocol) è uno degli esempi più noti di protocollo di livello di rete.**Immagine che contiene testo, Carattere, giallo

Descrizione generata automaticamente**

**Livello di Trasporto (Transport Layer):** Il livello di trasporto si concentra sulla trasmissione end-to-end dei dati tra i dispositivi. Fornisce servizi come la segmentazione dei dati in pacchetti più piccoli, il controllo degli errori e il controllo del flusso. Due protocolli chiave a questo livello sono **TCP** (Transmission Control Protocol) e **UDP** (User Datagram Protocol).



**Livello di Sessione (Session Layer):** Questo livello è responsabile dell'apertura, della gestione e della chiusura delle sessioni di comunicazione tra dispositivi. Aiuta a stabilire e mantenere il dialogo tra le applicazioni su entrambi i lati della comunicazione. Può includere funzionalità come il controllo della sincronizzazione e la gestione delle sessioni.



**Livello di Presentazione (Presentation Layer)**: Il livello di presentazione si occupa della traduzione, dell'elaborazione e della conversione dei dati in un formato comune comprensibile da entrambi i dispositivi. Questo livello può gestire la crittografia per la sicurezza dei dati, la compressione per ridurre la larghezza di banda necessaria e la conversione dei formati dati.



**Livello di Applicazione (Application Layer):** Questo è il livello più alto e fornisce servizi direttamente alle applicazioni utente. Include una vasta gamma di protocolli e servizi che consentono alle applicazioni di comunicare attraverso la rete. Esempi di protocolli di livello di applicazione includono HTTP per il web, SMTP per la posta elettronica, FTP per il trasferimento di file e molti altri.

## Tipologie di Connessione

**Connection-Oriented**: Si stabilisce la connessione, si stabilisce un percorso (instradamento), si effettua la comunicazione inviando i pacchetti ed infine si rilascia la connessione.

Il percorso è definito a priori.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page25image29452240page25image13294368page25image13290000 | | | |
|  |  | |  |

**Connectionless:** I pacchetti vengono inviati sulla rete senza un percorso predefinito e potrebbero arrivare in ordine sparso o perdersi.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page26image29478976page26image13177552page26image13173744 | | | |
|  |  | |  |

## Affidabilità di un servizio

**ACK:** Sistema di notifica di azioni tra hosts

**RELIABLE**: I dati devono essere tutti consegnati al destinatario e per ogni pacchetto ricevuto viene inviato un ack al mittente. E’ affidabile, ma lento.

**UNRELIABLE:** Non c’è nessuna garanzia che i dati vengano consegnati. E’ veloce, ma poco affidabile.

E’ possibile andare a combinare connessione ed affidabilità per ottenere servizi:

**Reliable Connection-Oriented:** Altamente affidabile

**Reliable Connectionless:** I dati saranno ricevuti tutti, ma non nell’ordine desiderato. Vengono utilizzati ACK.

Anche chiamato acknowledged datagram service

**Unreliable Connection-Oriented:** Possibili perdite di dati

**Unreliable Connectionless**: Altamente inaffidabile con possibili perdite di dati non ordinati. Anche chiamato datagram service

# **LEZIONE 3 – MODELLO TCP/IP E CONFRONTO**

## Modello TCP/IP

Il modello TCP/IP è un modello di comunicazione di rete che è stato sviluppato dal Dipartimento della Difesa degli Stati Uniti ed è ampiamente utilizzato per la progettazione, l'implementazione e la gestione delle reti di computer. È stato creato prima del modello OSI (Open Systems Interconnection) ed è ancora comunemente utilizzato come modello di riferimento per la comprensione delle reti di computer (**standard de facto**). Il modello TCP/IP contiene protocolli multipli oltre a quelli da cui prende il nome.

Il modello TCP/IP è diviso in quattro livelli, ciascuno dei quali svolge un ruolo specifico nella trasmissione dei dati attraverso una rete.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page3image29756880page3image15731328page3image15729424 | | | |
|  |  | |  |

Come si può notare, nel modello TCP/IP:

* Il livello **Applicazione** comprende il livello **Presentazione** e **Sessione** del modello ISO/OSI
* Viene ridefinito il livello **Rete** in livello **Internet**
* Il livello **Accesso Rete** comprende sia il livello **Collegamento** sia quello **Fisico**

## Livello Accesso Rete

In inglese **host-to-network.**

Questo è il livello più basso del modello e si occupa della trasmissione dei dati su un singolo collegamento fisico. Include protocolli e tecnologie per la gestione dell'accesso al mezzo di trasmissione, la rilevazione degli errori e la trasmissione affidabile dei dati all'interno dello stesso segmento di rete. Alcuni esempi di protocolli di livello di collegamento includono Ethernet e Wi-Fi.

Non specifica come questo debba funzionare nel dettaglio e lascia la possibilità di utilizzare I dispositive di rete con propri protocolli ( Nel caso di comunicazione con dispositivi differenti l’apparato di rete provvederà alla conversione multiprotocollo ).

## Livello Internet

Questo livello gestisce il routing dei dati tra diversi segmenti di rete. È responsabile della consegna dei pacchetti da un nodo a un altro attraverso una serie di reti interconnesse. Il protocollo principale di questo livello è l'Internet Protocol (IP), che assegna indirizzi IP unici a ciascun dispositivo nella rete e determina come i pacchetti di dati vengono instradati da un dispositivo all'altro.

## Livello Trasporto

Questo livello si occupa del trasferimento dei dati end-to-end tra dispositivi. È responsabile del controllo del flusso, della segmentazione dei dati in pacchetti più piccoli e della gestione degli errori. Due dei protocolli più noti a questo livello sono il Transmission Control Protocol (TCP), che fornisce una comunicazione affidabile, e il User Datagram Protocol (UDP), che è più veloce ma meno affidabile.

**TCP**:

* Reliable Connection-Oriented
* Mittente - Frammenta il messaggio in pacchetti
* Destinatario - I pacchetti vengono riuniti per ricostruire il messaggio

**UDP**:

* Unreliable Connectionless
* Perdita di pacchetti

## Livello Applicazione

Questo è il livello più alto e contiene le applicazioni che consentono agli utenti di interagire con la rete. Include una vasta gamma di protocolli e servizi, come HTTP per il web, SMTP/POP/IMAP per la posta elettronica, FTP per il trasferimento di file, DNS per il mapping degli indirizzi IP e molti altri. Questo livello si interfaccia direttamente con le applicazioni utente e fornisce loro l'accesso ai dati di rete.

Il modello TCP/IP nasce come modello di riferimento e non si fa troppa distinzione tra protocolli, servizi ed interfacce.

# **LEZIONE 4 – LIVELLO FISICO E TEORIA DEI SEGNALI**

## Basi della trasmissione

Un segnale sinusoidale è una forma d’onda periodica e regolare che si ripete in un periodo di tempo *t*

L'informazione può essere trasmessa a distanza variando opportunamente una qualche caratteristica fisica del segnale scelto per la trasmissione.

Il segnale, inoltre, si propaga attraverso un qualche mezzo di trasmissione con una certa velocità *t* per un intervallo di tempo 𝛥*t*

Il segnale ha le seguenti caratteristiche:

* **Ampiezza (A)**: Differenza fra il valore massimo ed il valore minimo
* **Periodo (T)**: Quantità di tempo prima della ripetizione del segnale
* **Frequenza (F)**: Numero di oscillazioni sul periodo T (1/T)
* **Fase (Φ)**: Scostamento dall’origine

Immagine che contiene diagramma, linea, Diagramma

Descrizione generata automaticamente

## Mezzi di Trasmissione

Il segnale può essere trasmesso principalmente attraverso tre diversi fenomeni fisici associati ad un determinato mezzo di trasmissione

• **Corrente elettrica** 🡪 Cavi in rame  
• **Luce** 🡪 Fibra ottica  
• **Onde elettromagnetiche** 🡪 Wireless (una combinazione di campo elettrico e campo magnetico variabili, che propagandosi nello spazio è in grado di indurre a distanza una corrente elettrica in un dispositivo ricevente (antenna).

## Tipologia di segnale

Il segnale si divide in due categorie:

* ***Analogico***: il valore del segnale può variare ***gradualmente*** in un intervallo costituito da un ***numero infinito*** di possibili valori;
* ***Digitale***: il segnale varia ***bruscamente*** assumendo in ogni istante solo uno di un ***insieme finito di valori.***

I fenomeni naturali sono tutti esclusivamente analogici.

A causa dell’interazione con il mezzo trasmissivo, la ***forma del segnale a destinazione non sarà mai esattamente quella di partenza***.

## Analisi Spettrale

### Teorema di Fourier

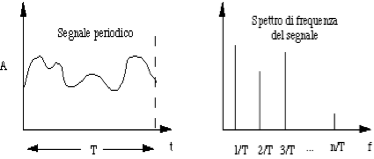
Una funzione **g(t),** definita in un intervallo finito T, può essere considerata come una funzione periodica di periodo T, e quindi espressa come somma di un numero infinito di funzioni sinusoidali

Immagine che contiene Carattere, calligrafia, testo, linea

Descrizione generata automaticamente

* Tale funzione può essere vista anche in base alla frequenza fondamentale **f = 1/T**
* **an** e **bn** sono le ampiezze dell’ennesima armonica che ha frequenza **n∗f**
* ***c*** è detto coefficiente di Fourier

Un qualunque segnale ***g(t)***, di durata T, può essere rappresentato dal suo ***spettro di frequenze*** (la sua scomposizione in sinusoidi).



* L’analisi spettrale porta a vedere il segnale rappresentato da un insieme di frequenze
* Queste frequenze rappresentano le sue sinusoidi
* La scomposizione in frequenze è detto spettro di frequenze
* L’intervallo delle frequenze di tutte le sinusoidi è detto **frequency band**
* Minore è la durata del periodo T, maggiore sarà il valore della frequenza fondamentale
* Quanto più velocemente varia g(t), tanto più numeroso sono le armoniche che la descrivono

Dato un mezzo fisico di comunicazione abbiamo che le seguenti definizioni:

* **Banda passante** - intervallo di frequenze che il mezzo fisico è in grado di trasmettere senza alterarle oltre certi limiti
* **Attenuazione e Ritardo** - Alterazioni del segnale. L’attenuazione dipende dalla frequenza del segnale ed è proporzionale alla distanza percorsa
* **Ampiezza di banda** - quantità di informazioni che è possibile trasmettere in un determinato lasso di tempo. (Possono esserci delle attenuazioni sulle bande tramite l’utilizzo di filtri (es. passa-basso, passa-banda)

***Nella trasmissione in un mezzo trasmissivo***:

l*’****attenuazione*** *subita dal segnale* ***dipende dalla frequenza*** *del segnale* ***ed è proporzionale alla distanza percorsa****;*

*se la* ***banda passante*** *è inferiore alla* ***banda di frequenza****, il segnale viene* ***distorto*** *(****privato di alcune armoniche****).*

***Se un numero sufficiente di armoniche arriva a destinazione, il segnale è comunque utilizzabile****.*

Un segnale, anche se distorto, può essere ricostruito se un numero sufficiente di armoniche giunge a destinazione

### Teorema di Nyquist

Un segnale analogico di banda h può essere completamente ricostruito mediante una campionatura effettuata 2 ∗ h volte al secondo

***In ambito binario***: Se ciascun campione può assumere uno di ***n*** valori distinti, ***il segnale risulta completamente rappresentato*** con ***2\*h\*(lg n) bit per ogni secondo (lg 🡪 log in base 2)***

Tale valore è chiamato anche **bit rate**

***Baud rate:*** *Velocità di comunicazione (numero di volte in un secondo in cui è possibile cambiare il valore)*

### Rapporto Segnale/Rumore

Non esiste un canale di comunicazione perfetto, per cui è sempre possibile che il segnale sia distorto.

Si utilizza il rapporto segnale/rumore (signal to noise ratio)

SNR = 10 ∗ log(S/N) (log 🡪 log in base 10) e si misura in decibe (db)

|  |  |
| --- | --- |
| Rapporto S/N | Decibel |
| 2 | 3 |
| 10 | 10 |
| 100 | 20 |
| 1000 | 30 |

*Il teorema di Nyquist è valido per canali totalmente privi di disturbi* (purtroppo non è realistico).

### Teorema di Shannon

il massimo data rate di un canale rumoroso, con banda passante di ***h Hz*** e rapporto segnale/rumore pari a ***S/N***, è dato da: ***massimo data rate (bit/sec)= h\*lg2(1+S/N)***

Su un canale con banda ***3k Hz e S/N=30dB*** (tipici di una normale linea telefonica) massimo si può arrivare a ***30.000 bps.***

***In generale***:

* più alto è il numero di ***bps*** da trasmettere, più ampia deve essere la banda passante;
* a parità di mezzo utilizzato, minore è la lunghezza del canale di trasmissione tanto più è alto il numero di ***bps*** che si possono trasmettere (attenuazioni e sfasamenti restano accettabili);
* la trasmissione digitale è più critica di quella analogica (genera frequenze più alte), ma può essere più facilmente "rigenerata" lungo il percorso (in quella analogica ogni amplificazione introduce distorsione, che si somma a quella degli stadi precedenti).

## Trasmissione

Trasmissione **baseband** (trasmissione in banda base): la singola trasmissione impegna l'intera banda passante.

Trasmissione **broadband**: attraverso opportune tecniche di multiplazione si effettuano contemporaneamente più trasmissioni distinte (nella telefonia con broadband si indicano trasmissioni che impegnano più di 4 kHz)

## Mezzi trasmissivi

**Doppino intrecciato**: coppia di conduttori in rame intrecciati in forma elicoidale (**l’intreccio riduce i disturbi causati dalle interferenze)**

* usato, in particolare, per le connessioni terminali del sistema telefonico (da casa dell’utente alla centrale più vicina).
* larghezza di banda - dipende dalla distanza (100 -1000 Mb/s a 100 metri).

**Cavo coassiale**: conduttore centrale in rame circondato da uno strato isolante all'esterno del quale vi è una calza metallica, il tutto rivestito in plastica. Il miglior isolamento, rispetto al doppino, consente maggiori velocità di trasmissione e distanze superiori.

Immagine che contiene diagramma, schizzo, cerchio, testo

Descrizione generata automaticamente

* Usato in passato nel sistema telefonico per le tratte a lunga distanza, è ormai sostituito quasi ovunque dalla fibra ottica (oggi è usato per la TV via cavo ed in vecchie LAN)

***Fibra ottica***: sottilissimo cilindro centrale in vetro (***core***), circondato da uno strato esterno (***cladding***) di vetro avente un diverso indice di rifrazione e da una guaina protettiva.

In genere più coppie sono contenute insieme in una stessa guaina esterna.

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page19image29606512page19image16029040page19image16033744 | | | |
|  |  | |  |

Un sistema di trasmissione ottico ha tre componenti

1. **sorgente luminosa** 
   1. converte un segnale elettrico in impulsi luminosi (può essere un LED o un laser)
2. **mezzo di trasmissione**
   1. fibra ottica
3. **fotodiodo ricevitore** 
   1. converte gli impulsi luminosi in segnali elettrici (l’ordine dei Gbps deriva dal tempo di risposta dei fotodiodi che è dell’ordine del nsec)

***Quando un raggio di luce attraversa il confine fra il core ed il cladding subisce una deviazione.***

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page20image29562560page20image13285712page20image13276304 | | | |
|  |  | |  |

La deviazione dipende dagli ***indici di rifrazione*** dei due materiali ed è tale che ***per certi angoli di incidenza***, ***il raggio resta intrappolato all'interno del core***.

Le fibre ottiche si dividono in:

* ***multimodali***: (**core di 50 micron**) raggi diversi con diversi angoli (***mode***) possono contemporaneamente propagarsi nella stessa fibra;
* ***monomodali***: (**core di 8-10 micron**) la luce di un singolo raggio avanza nella fibra che si comporta come una ***guida d'onda***. Sono più costose ma garantiscono distanze maggiori (fino a 30 km).

Nelle fibre ottiche un impulso luminoso rappresenta un **1**, mentre la sua assenza rappresenta uno 0.

* Le attuali fibre consentirebbero velocità di trasmissione di 50 Tbps ad un bassissimo tasso d’errore
* Mancando al momento sistemi di conversione luminoso/elettrico in grado di operare a tali velocità, la pratica corrente limita l’uso delle fibre a qualche Gbps

La ***bassa attenuazione*** nella trasmissione in fibra è dovuta:

* alla ***particolare trasparenza*** delle fibre ottiche (se il mare fosse fatto di questo vetro si vedrebbe il fondo...);
* all’utilizzo di ***tre particolari bande (finestre)*** per la trasmissione (tutte tra ***infrarosso e UV***: 0.85μ, 1.30μ, 1.55μ –perdita di meno del 5% per Km-), larghe da 25.000 GHz a 30.000 Ghz ciascuna.

## Topologie di reti in fibra ottica

**Anello**: concatenando più fibre ottiche si crea un anello. L'interfaccia del singolo sistema può essere passiva (fa passare l'impulso luminoso nell'anello) o attiva (converte l'impulso luminoso in elettrico, lo amplifica e lo riconverte in luce);

**Stella passiva**: l'impulso, inviato da un trasmettitore, arriva in un cilindro di vetro al quale sono attaccate (fuse) tutte le fibre ottiche. Si realizza così una rete broadcast.

***Vantaggi*** delle fibre ottiche rispetto al rame:

* **banda**:*due fibre sono più capaci di 1000 doppini*;
* **peso**: *100 kg/km contro 8.000 kg/km*;
* ***totale insensibilità a disturbi elettromagnetici***;
* ***difficili intrusioni***.

**Svantaggi** delle fibre ottiche rispetto al rame:

* ***costo delle giunzioni***;
* ***comunicazione unidirezionale***

## Trasmissioni wireless

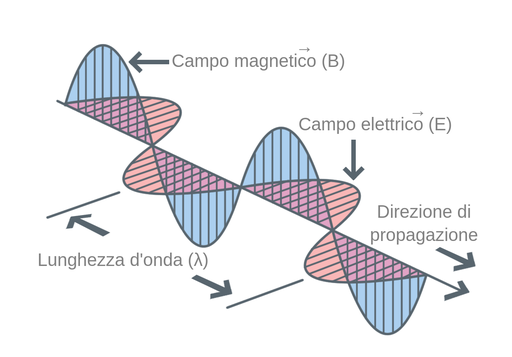
Si utilizzano le onde elettromagnetiche che viaggiano nel vuoto alla velocità della luce (c=3\*10^8 m/s).

In particolare:

Un’onda elettromagnetica monocromatica (cioè con una ben definita frequenza e lunghezza d’onda) è costituita da un campo elettrico (E) e un campo magnetico (B) mutuamente perpendicolari che oscillano in fase fra loro perpendicolarmente alla direzione di propagazione

**Caratteristiche di un’onda elettromagnetica:**

* **Frequenza f: numero di oscillazioni al secondo. Si misura in Hertz (Hz)**
* **Lunghezza d’onda λ: distanza tra due massimi o minimi consecutivi.**
* **La velocità c è data dalla relazione c = f ∗ λ**



La lunghezza d’onda decresce al crescere della frequenza. λ=c/f

## Classificazione onde elettromagnetiche

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| page32image53433840page32image35573584page32image35571568 | | | |
|  |  | |  |

* **Onde radio a bassa frequenza**

Sono le onde VLF, LF, MF.

Si propagano in tutte le direzioni e passano attraverso gli edifici percorrendo lunghe distanze.

Sono soggette ad interferenze elettromagnetiche.

* **Onde radio ad alta frequenza**

Sono le onde HF, VHF.

Con queste frequenze inizia ad aversi una propagazione in linea retta con lambda (lungh. Onda) dell’ordine del cm/mm.

Rimbalzano gli ostacoli e tendono essere assorbite dal suolo.

Vengono impiegate per le trasmissioni direzionali, le onde che colpiscono la ionosfera vengono rifratte e rispedite sulla terra.

Sono soggette ad interferenze elettromagnetiche.

* **Microonde**

Sono onde con frequenza intorno ai 100Mhz e si propagano in linea retta.

A causa della curvatura della terra, la trasmissione oltre certe distanze richiede l’uso di ripetitori (la distanza fra i ripetitori cresce come la radice quadrata dell’altezza delle torri: per torri alte 100 m i ripetitori possono essere distanti 80 Km).

Usate per trasmissioni a grande distanza e come bande per applicazioni industriali-scientifiche-mediche.

A causa della dispersione nello spazio alcune onde possono essere rifratte, e quindi arrivare in ritardo e fuori fase (Multipath Fading: distruzione del segnale causata dall’arrivo fuori fase di onde rifratte).

Per bande fino a circa 8 Ghz, le onde vengono assorbite dall’acqua (pioggia)

* **Onde infrarosse e millimetriche**

Usate per comunicazioni su piccole distanze (telecomandi televisori)

Sono relativamente direzionali.

Non passano attraverso corpi solidi

Usate in ambienti chiusi

Non interferiscono con quelle impiegate nei locali attigui (usando il proprio telecomando non si cambia il canale al televisore del vicino).

## Trasmissione e modulazione

Le informazioni si trasmettono tramite modulazione

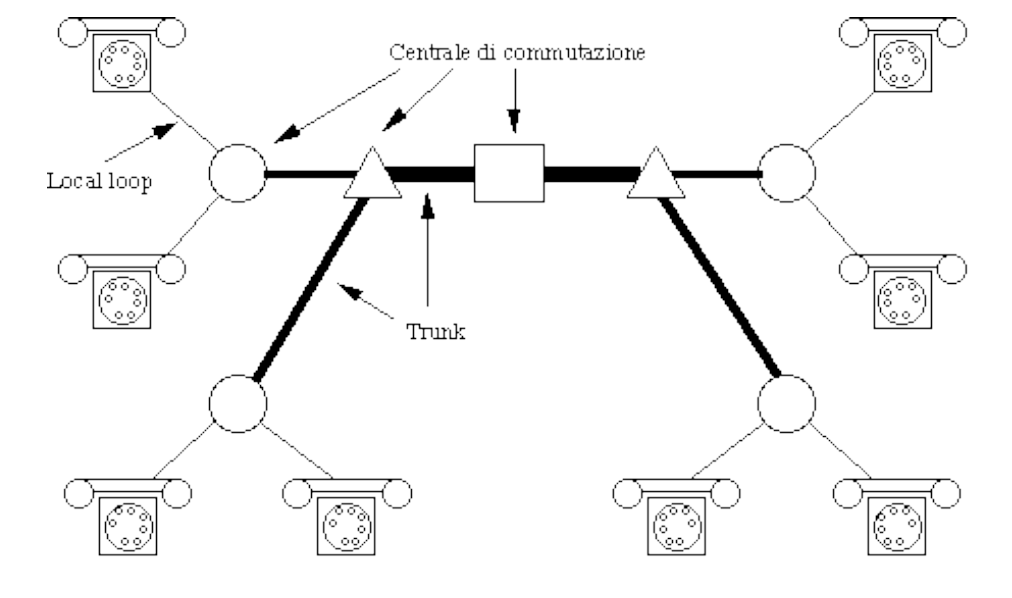
Si possono modulare

* Ampiezza (A)
* Frequenza (f )
* Fase (Φ)

# **LEZIONE 5 - SISTEMA TELEFONICO E MODULAZIONE**

Il sistema telefonico (**rete pubblica telefonica commutata)** è nato ed evolutosi per la fonia, poco adatto per la trasmissione dati

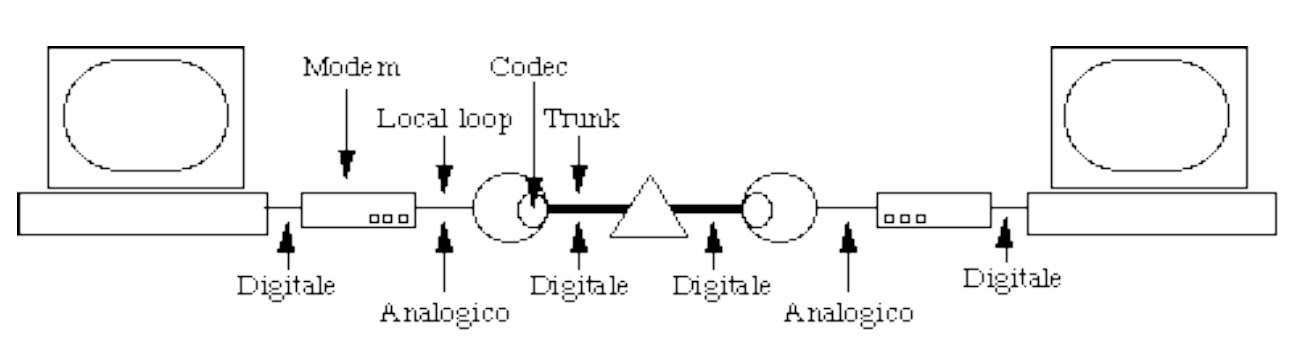
I sistemi telefonici attuali seguono una **gerarchia multilivello** con elevata ridondanza



**Local loop:** collega il telefono alla più vicina **centrale di commutazione**. Il collegamento è realizzato principalmente da un **doppino** lungo da 1 a 10km e trasporta un segnale analogico che occupa una **banda di 3 kHz 🡪** detto anche **ultimo miglio**

**Centrali di commutazioni:** quasi sempre digitali perché, oltre a data rate più alti, è più facile sia ricostruire il segnale senza introdurre errori e mescolare voce, dati, video.

**Trunk:** collegano le **centrali di commutazioni**. I collegamenti sono realizzati tipicamente in **fibra ottica**



**Modem:** effettua la trasformazione del segnale **digitale/analogico** in trasmissione e **analogico/digitale** in ricezione, necessarie per trasmettere dati sul **local loop** a 3 kHz

**Codec:** effettua le trasformazioni **analogico/digitale** nella prima centralina di commutazione e **digitale/analogico** nell’ultima

Le trasmissioni sui **local loop** sono soggette ad **attenuazione e distorsione.**

Poiché l’ampio spettro di frequenze generato dalle onde quadre contrasta con la banda ridotta del local loop, **la trasmissione digitale sui local loop può avvenire solo a bassissime velocità.**

Per trasmettere un segnale digitale sul local loop, si modula un segnale sinusoidale, detto **portante**, la cui frequenza è compresa tra 1 e 2 kHz (la banda del local loop è di 3 kHz)

Un segnale può essere modulato in:

* **Ampiezza 🡪** si varia l’ampiezza
* **Frequenza 🡪** si varia la frequenza
* **Fase 🡪** si varia la fase (cioè lo scostamento rispetto al segnale originale)

Un ulteriore aumento delle prestazioni si può ottenere effettuando una **compressione dei dati** prima di trasmetterli.

Per la compressione dei dati, esistono due standard: **V.42 bis (de iure)** emesso da ITU e **MNP 5** emesso da Microom Network (**de facto)**

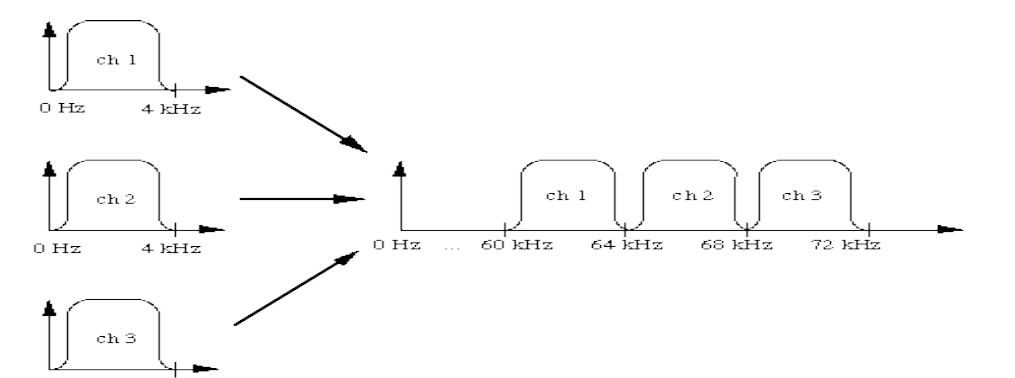
Maggiori velocità sono oggi possibili con nuove tecnlogie quali ISDN, ADSL, ATM…

I trunk trasmettono migliaia di informazioni simultaneamente.

La capacità di trasmettere migliaia di informazioni simultaneamente prende il nome di **multiplexing TDM e FDM.**

* **Frequency Division Multiplexing (FDM):** adatto alla gestione di segnali analogici

Lo spettro di frequenza è suddiviso in bande più piccole, e ogni comunicazione ha l’esclusivo uso di una di esse.



* **Time Division Multiplexing (TDM):** adatto per la gestione di dati in forma digitale.

i bit provenienti da diverse connessioni vengono a turno prelevati ed inviati su un'unica

connessione ad alta velocità:

Questa tecnica con il tempo si è confermata essere molto affidabile se non fosse per il fatto che l’ITU non riuscì a creare uno standard internazionale facendo sì che nel mondo venissero utilizzare una serie di modalità incompatibili tra di loro.

Nord America e Giappone usano la portante ***T1***, essa è composta da 24 canali vocali uniti in multiplexing, ognuno dei quali, a turno, inserisce 8 bit nel flusso in uscita.

Al di fuori del Nord America e Giappone, al posto della T1 viene usata la portante **E1**, composta da 32 canali (30 dati e 2 controllo) con valori ad 8 bit ogni 125 ms.

Con il passare degli anni, per creare collegamenti sempre più performanti si è deciso di:

• ***T2***: collegare 4 canali T1, ognuno dei quali garantiva 1.544 Mbps, per formare un unico canale che garantisse velocità di 6,312 Mbps (non 6,176 perché c’è l’aggiunta di bit di controllo).

• ***T3***: collegare 7 canali T2, ognuno dei quali garantiva 6,312 Mbps, per formare un unico canale che garantisse velocità di 44,736 Mbps.

• ***T4***: collegare 6 canali T3, ognuno dei quali garantiva 44,736 Mbps, per formare un unico canale che garantisse velocità di 274,176 Mbps.

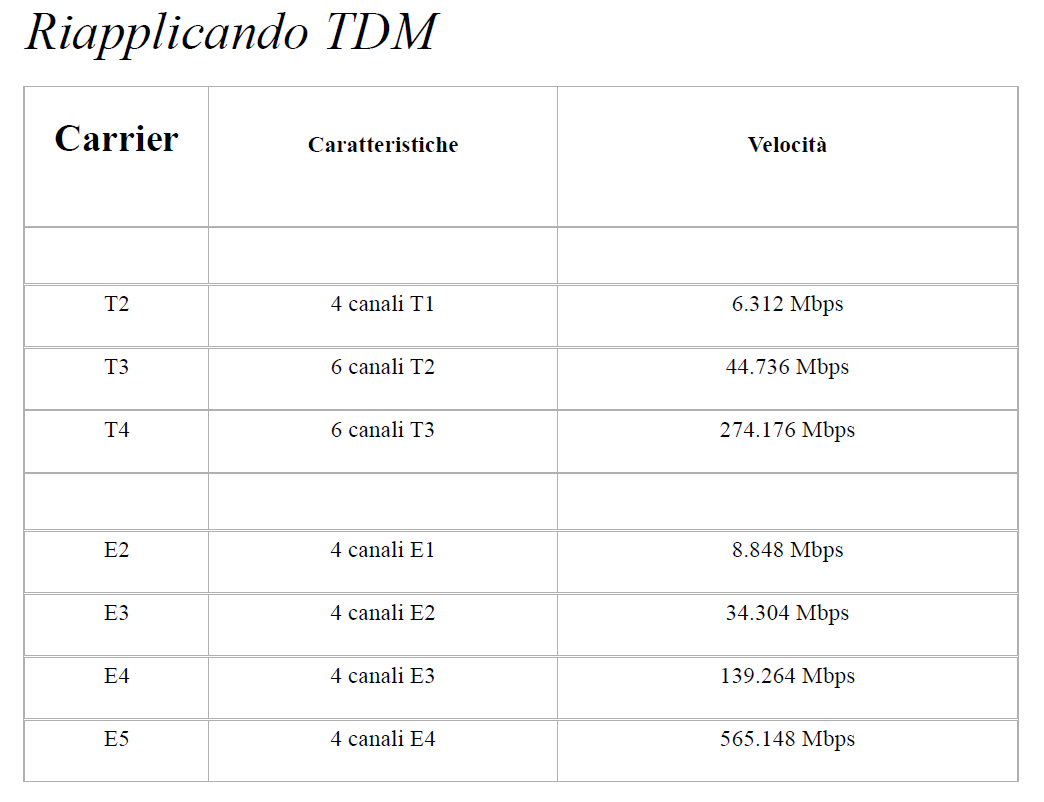
La differenza tra i sistemi richiede costose apparecchiature di conversione ai confini fra un l’uno

e l'altro.

* **SDH** (Synchronous Digital Hierarchy): gerarchia unificata a livello mondiale, introdotta dal CCITT

verso la metà degli anni '80.

* **SONET** (Synchronous Optical NETwork): nato negli USA, si basa sul canale STS-1 (810 byte ogni 125 μs: 51.84 Mbps) non presente in SDH.



Poiché IL **TDM** può essere usato solo nei segnali digitali e l’ultimo miglio (local loop) produce segnali analogici, è necessario fare una conversione nella centrale locale, dove i segnali vengono raggruppati per essere spediti lungo i **trunk**

I segnali analogici sono digitalizzati da un dispositivo chiamato **codec.** Il codec estrae 8000 campioni al secondo (uno ogni 125 ms) perché secondo il teorema di Nyquist sono sufficienti a catturare tutta l’informazione da un canale telefonico a 4 kHz. Ad un tasso di campionamento più basso l’informazione andrebbe persa, viceversa, non si otterrebbe alcuna informazione aggiuntiva.

Questa tecnica è chiamata **PCM** (**P**ulse **C**ode **M**odulation, modulazione codificata di impulsi) ed è il cuore dei sistemi telefonici moderni. Di conseguenza, quasi tutti gli intervalli di tempo all’interno di un sistema telefonico sono multipli di 125ms.

All’altro capo della linea il segnale verrà riconvertito in analogico dal codec anche se non sarà del tutto uguale a quello di partenza.

Finora abbiamo analizzato il sistema telefonico dal punto di vista dell’impianto esterno (ultimo miglio e i trunk), andiamo ora ad analizzare l’impianto interno, quello composto quindi dai commutatori. Attualmente le reti usano due tecniche di commutazione:

• ***Commutazione di circuito***: tecnica su cui si basa il sistema telefonico tradizionale, tale tecnica, concettualmente, quando una persona avvia una telefonata, l’apparecchiatura di commutazione cerca di creare un percorso fisico completo tra il chiamante ed il chiamato. Agli inizi della telefonia era l’operatore manualmente a creare questo percorso oggi ci sono dispositivi di commutazione automatica.

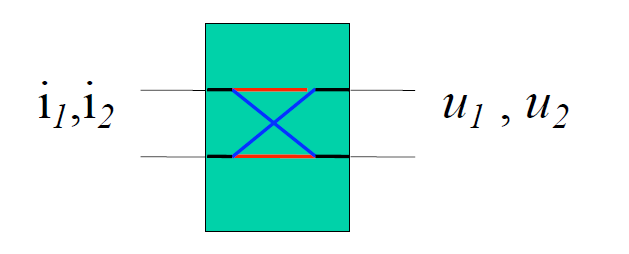
• ***Commutazione di pacchetto***: con questa tecnica la comunicazione tra i due utenti avviene mediante l’invio di pacchetti, questo fa si che si hanno notevoli benefici, tra cui, non avere la necessità di un percorso fisico dedicato, poiché ogni pacchetto segue una strada diversa, seguendo la disponibilità della banda dei percorsi disponibili, inoltre il pacchetto appena generato viene inviato, senza la necessità di creare prima un percorso. Di seguito vediamo le differenze principali tra le due tecniche.

I dispositivi di commutazione sono:

* **Switch**

dispositivo a semiconduttori con due ingressi (i1, i2), due uscite (u1, u2) e due possibili

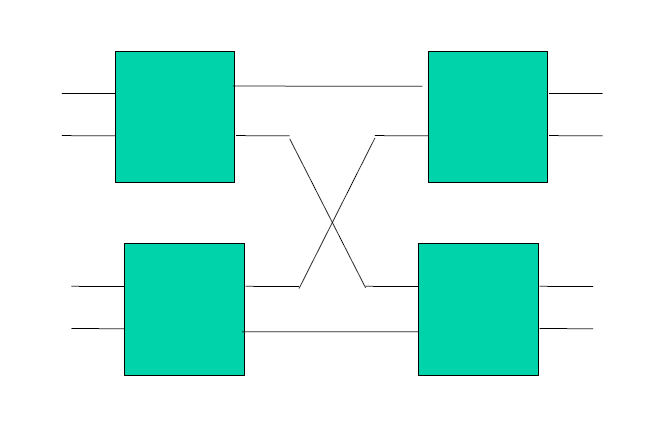
stati:



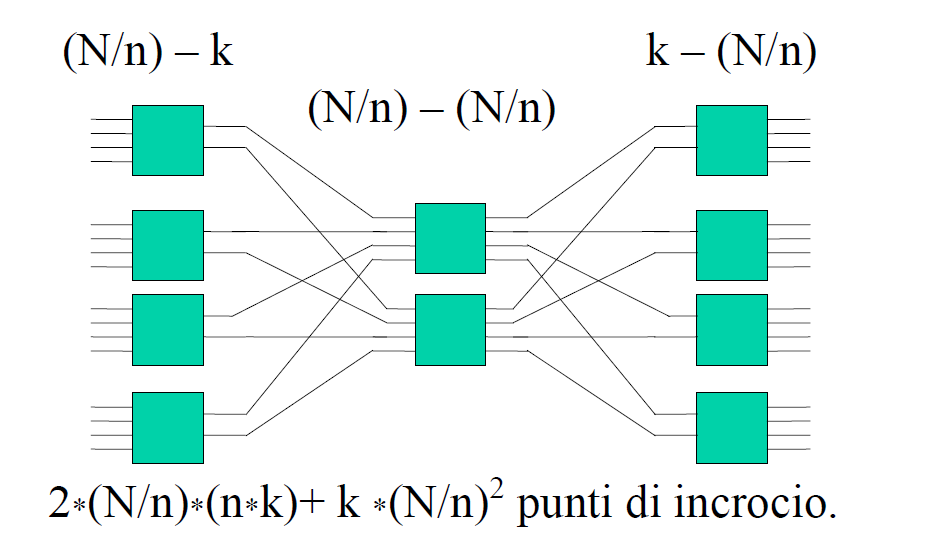
*i1* è connesso a *u1* e *i2* è connesso a *u2*;

*i1* è connesso a *u2* e *i2* è connesso a *u1*.

* **Crossbar**: N ingressi, N uscite ed N2 punti di incrocio, realizzati ciascuno con uno switch.



* **Switch multilivello**: (multistage switch): utilizzo di più crossbar di piccole dimensioni (ad es. 2\*2), organizzati in gruppi su più livelli successivi



# **LEZIONE 6 – IL LIVELLO DATA LINK**

## Introduzione

Nel livello data link discuteremo degli algoritmi usai per ottenere una comunicazione affidabile ed

efficiente tra unità d’informazione chiamate frame fra due macchine adiacenti, ovvero due

macchine collegate da un canale di comunicazione che agisce concettualmente come un cavo.

## Progettazione del livello Data Link

Il livello Data Link fa uso del servizio messo a disposizione dal livello fisico per inviare e ricevere bit su un canale di comunicazione.

Ha molte funzioni, tra cui:

* Fornire un’interfaccia di servizio ben definita per il livello tre
* Gestire gli errori di trasmissione
* Regolare il flusso dati in modo che i buffer dei dispositivi lenti non vengano inondati dai trasmittenti veloci

Il livello Data Link può essere progettato in modo da offrire una varietà di servizi che cambiano da protocollo a protocollo, come ad esempio:

* Servizio senza conferma (**unacknowledged**) senza connessione
* Servizio con conferma (**acknowledged)** senza connessione
* Servizio con conferma orientato alla connessione

Il servizio di tipo ***unacknowledged*** senza connessione consiste nell’avere una macchina sorgente che invia i frame senza preoccuparsi dell’arrivo effettivo di quest’ultimi.

In particolare, il destinatario non invierà al mittente una conferma.

Se un frame viene perso a causa del rumore sulla linea, non viene fatto alcun tentativo di rilevare la perdita o di correggerla nel livello data link. L’uso di questa classe di servizio è appropriato quando la frequenza degli errori di trasmissione è molto bassa oppure quando si deve utilizzare un traffico real-time, come la trasmissione vocale, per le quali gli effetti del ritardo sono peggiori di quelli derivanti dalla trasmissione di dati sbagliati.

Il servizio ***acknowledged*** senza connessione, continua a non utilizzare alcun tipo di connessione logica, però quando il ricevente riceve un frame, spedisce al mittente una conferma chiamata **acknowledgment (ACK).** In questo modo il mittente riesce a sapere se il frame è arrivato a destinazione in modo corretto oppure no. Se non è arrivato in uno specifico intervallo temporale, il frame può essere rispedito. Questa classe di servizio è utilizzata per i canali di trasmissione poco affidabili come ad esempio il WI-FI.

L’ACK è da considerarsi come un’ottimizzazione e mai come un requisito obbligatorio.

Il servizio ***acknowledged orientato alla connessione*** è un tipo di servizio dove il mittente e destinatario stabiliscono una connessione prima di iniziare a trasferire i dati.

Ogni frame trasferito è numerato e il livello data link garantisce che venga effettivamente ricevuto una sola volta e nell’ordine corretto, quindi si tratta di un servizio che consente il trasferimento affidabile di un flusso di bit.

E’ utilizzato per collegamenti lunghi e inaffidabili come i canali satellitari e linee telefoniche a lunga distanza.

Se, in queste situazioni, si utilizzasse semplicemente un servizio con ACK senza connessione, è possibile che i frame di ACK vadano persi e ciò causerebbe un re-invio di frame duplicati che andrebbe solamente a sprecare la banda del canale di comunicazione.

## Operazioni del livello due

In trasmissione:

* suddivide il flusso di bit, che arriva dal livello tre, in una serie di frame;
* calcola una funzione (**checksum**) per ciascun frame;
* inserisce il checksum nel frame;
* consegna il frame al livello uno, il quale lo spedirà come sequenza di bit.

In ricezione:

* riceve una sequenza di bit dal livello uno;
* ricostruisce da essa un frame dopo l'altro;
* per ciascun frame ricalcola il checksum;
* se il checksum ricalcolato è diverso da quello contenuto nel frame, il frame viene scartato.

## Suddivisione in frame

L’approccio del livello data link è quello di suddividere il flusso di bit in una serie discreta di frame, calcolare un valore a lunghezza fissa, chiamato **checksum**, per ogni frame e includere il checksum nel frame quando viene trasmesso. Quando un frame arriva a destinazione, il checksum viene ricalcolato e se differisce, il livello data link sa che c’è stato un errore e prende provvedimenti, per esempio scartando il frame o restituendo un messaggio di errore.

Suddividere il flusso di bit in frame è un’operazione che viene chiamata **framing** ed è possibile effettuarla attraverso 4 metodi:

* **Conteggio dei byte**
* **Flag byte con byte stuffing**
* **Flag bit con bit stuffing**
* **Violazioni della codifica del livello fisico**

Il metodo **conteggio dei byte** consiste nell’usare un campo nell’header per specificare il numero di byte del frame. Quando il livello data link della destinazione legge tale numero, sa quanti byte seguiranno e quindi dove si troverà la fine del frame.

Tuttavia questo metodo non è efficiente, poiché l’informazione viaggia nel frame che è soggetto a rumore e quindi il valore contenuto nell’header potrebbe cambiare, ottenendo letture di frame inconsistenti.

Il secondo metodo chiamato **Flag byte con byte stuffing** consiste nell’inserire byte speciali all’inizio e alla fine di ogni frame. La maggior parte dei protocolli utilizza un byte speciale che prende il nome di **flag byte (DLE – Data Link Escape)** , per delimitare sia l’inizio che la fine dei frame.

Quindi, due flag byte consecutivi indicano la fine e l’inizio di un frame.

**DLE STX** (Start of TeXt): inizio frame.

**DLE ETX** (End of TeXt): fine frame.

Questo metodo presenta tuttavia un problema, in quanto può accadere che nella trasmissione dati compaia naturalmente il flag byte, specialmente in quelli binari, quali immagini, video, audio ecc, interferendo così con le operazioni di framing.

La soluzione è quella di inserire un byte di escape (**un altro DLE)** subito prima di ogni occorrenza nei dati.

Il livello data link di destinazione provvederà a rimuovere i byte di escape prima di passarli e livello rete. Questa tecnica è chiamata **byte stuffing**

Il terzo metodo aggira il limite imposto dal byte stuffing di usare byte ad 8 bit. L’operazione di framing può essere effettuata anche a livello di bit in modo che i frame possano contenere un numero arbitrario di bit.

Ad esempio, ogni volta che si incontrano 5 bit **consecutivi** ad 1, si inserisce uno 0 aggiuntivo per delimitare il frame.

Anche in questo caso è presente lo stuffing.

L’ultimo metodo sfrutta il concetto che il livello fisico per conservare l’integrità dei dati inserisce ridondanza, quindi nell’invio di 4 bit di dati spesso si usano pacchetti da 5 bit, tenendo così 16 delle 32 combinazioni libere, questa tecnica usa proprio quelle combinazioni libere per inserire l’inizio e la fine del frame.

## Rilevazione e correzione degli errori

Molti fattori possono provocare errori, soprattutto sul local loop e nelle trasmissioni wireless (mentre sono piuttosto rari nei mezzi più moderni quali le fibre ottiche).

Gli errori sono dovuti in generale a **rumore di fondo**, **disturbi improvvisi** (ad es. fulmini), **interferenze** (ad es. motori elettrici).

Per trattare questi errori il livello data link aggiunge informazione ridondanti ai dati che devono essere spediti.

Una strategia consiste nell’includere informazioni ridondanti in un numero sufficiente da permettere al destinatario di dedurre i dati realmente trasmessi, l’altra consiste nell’introdurre una ridondanza di informazioni per permettere al destinatario di dedurre che c’è stato un errore (ma non quale).

La prima strategia usa **codici a correzione d’errore** (*error-detecting code*), l’altra utilizza **codici a rilevazione d’errore** (*error-detecting code).* L’uso di codici a correzione d’errore viene spesso indicato come **FEC** (*forward error correction – correzione d’errore in anticipo)*

## Codici a correzione di errore

Tra i codici a correzione d’errore troviamo il **codice di Hamming**.

Un frame (a parte i delimitatori) sarà composto di

**m = n + r** bit dove **m = bit del messaggio vero e proprio, r = redundant bit o check bit 🡪 bit per il controllo dell’errore.**

Per introdurre il codice di Hamming è importante stabilire cos’è la **distanza di Hamming.**

La **distanza di Hamming** è semplicemente il numero di bit in cui differiscono due configurazioni binarie. Chiameremo ognuna di queste configurazioni **Codeword (parola codice 🡪 sequenza di m+r bit)** e l’insieme di tutte le configurazioni **Codice (code 🡪 insieme prefissato di codeword)**

Nell’ambito tecnico, tale distanza è calcolata attraverso un’operazione bitwise (bit a bit) di OR esclusivo (**XOR)** e poi si contano i bit ad 1 di tale sequenza.

**Es:** D(**1**00**0**1 . **0**000**1**1) = 2

Quando consideriamo l’insieme di tutte le codeword, quindi il codice, possiamo stabilire il concetto di **distanza minima di Hamming** di un codice, ovvero la minor distanza considerando due sue parole qualsiasi.

La distanza tra le parole di un codice gioca un ruolo fondamentale nel meccanismo di rilevazione degli errori. Un codice più è ridondante e più è efficace per rilevare errori. Se un codice non è ridondante, non ci sono possibilità di rilevazione degli errori.

ES: Supponiamo di avere la seguente codifica:

A 000

B 001

C 010

D 011

E 100

F 101

G 110

H 111

Se vogliamo inviare A 🡪 000 , ma arriva B 🡪 001 , quindi c’è stato un errore sul singolo bit finale, **il ricevente non è in grado di rilevare se c’è stato un errore poiché la sequenza 001 è una codeword valida.**

Se invece intervallassimo le parole codice come nel seguente caso:

A 000

001

010

100

B 011

111

101

110

C ….

Qualunque bit corrotto in una codifica valida porterebbe ad un errore poiché la distanza di Hamming di questo codice è 2.

In generale, un codice per la rilevazione di modifiche **su** **d bit,** deve avere almeno **DMIN = d+1**

Di conseguenza se ho un codice con **DMIN=2**, sarò in grado di rilevare errori sul singolo bit e così via.

I codici di questo tipo sono detti **SEDC (Single Error Detection Code)** ed hanno tutti **DMIN** almeno pari a 2.

Hamming, ha dato anche un ulteriore apporto per poter, non solo rilevare l’errore, ma anche individuare dove è presente l’errore e correggerlo.

Questo avviene aggiungendo un numero di bit sufficienti di parità e ciascun bit di parità sarà la parità di alcuni bit e l’algoritmo sarà in grado di determinare anche se un bit di parità si è sporcato.

E’ importante notare che il numero di bit di controllo va scelto secondo la seguente formula:

Dove **r** sono i bit di controllo ed **m** i bit del messaggio originale.

Dato un messaggio M da inviare, il procedimento da applicare per ottenere il frame MS effettivamente trasmesso, consiste nel sistemare i bit su di una griglia di 12 posizioni, dove le posizioni in cui l'indice è potenza di 2 sono occupate dai bit di controllo, mentre tutte le altre

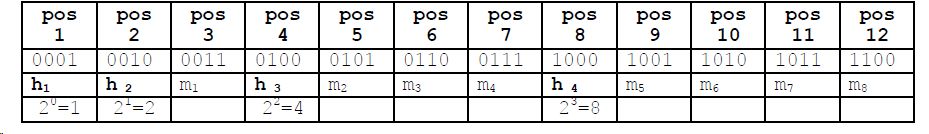
ospitano ordinatamente i bit del messaggio M.

Sia M = 10110110

con:

m1 = 1 m2 = 0 m3 = 1 m4 = 1 m5 = 0 m6 = 1 m7 = 1 m8 = 0

La stringa da trasmettere sarà la seguente:



Osservando la figura si può notare che per ogni byte da spedire il codice di Hammnig utilizza 4 bit di controllo: **h1 h2 h3 h4** nelle posizioni 1,2, 4 e 8; il codice è quindi completato dai bit del messaggio.

Nella seconda riga della stessa figura sono mostrati i valori binari delle posizioni occupabili dai bit; si noti che tutte le posizioni occupate dai bit controllori hanno un solo 1 in tale valore binario, mentre tutte le posizioni occupate dai bit controllati, i bit del messaggio, hanno almeno due 1.

Vediamo ora come si determinano i valori dei bit di controllo.

Il bit h1 si trova in posizione 1 (valore binario 000**1**) della stringa del codice Hamming. **Esso controlla tutti i bit che nel valore binario della posizione occupata hanno un 1 in quarta posizione incominciando da sinistra**, ovvero:

h1 controlla:

m1, posizione 001**1**

m2, posizione 010**1**

m4, posizione 011**1**

m5, posizione 100**1**

m7, posizione 101**1**

Il bit h2 si trova in posizione 2 (valore binario 00**1**0) della stringa del codice Hamming. **Esso controlla tutti i bit che nel valore binario della posizione occupata hanno un 1 in terza posizione incominciando da sinistra**, ovvero:

h2 controlla

m1, posizione 00**1**1

m3, posizione 01**1**0

m4, posizione 01**1**1

m6, posizione 10**1**0

m7, posizione 10**1**1

Il bit h3 si trova in posizione 4 (valore binario 0**1**00) della stringa del codice Hamming. **Esso controlla tutti i bit che nel valore binario della posizione occupata hanno un 1 in seconda posizione incominciando da sinistra**, ovvero:

h3 controlla

m2, posizione 0**1**01

m3, posizione 0**1**10

m4, posizione 0**1**11

m8, posizione 1**1**00

Il bit h4 si trova in posizione 8 (valore binario **1**000) della stringa del codice Hamming. **Esso controlla tutti i bit che nel valore binario della posizione occupata hanno un 1 in prima posizione incominciando da sinistra**, ovvero:

h4 controlla m5, posizione **1**001

m6, posizione **1**010

m7, posizione **1**011

m8, posizione **1**100

Nel codice così costruito ogni bit del messaggio da spedire, M, è controllato dal almeno due bit di controllo.

**I valori dei bit di controllo si ottengono mediante un’operazione di XOR tra tutti i bit controllati**, per cui avremo che:

h1 = m1 ⊕m2 ⊕m4 ⊕m5 ⊕m7 = 1 ⊕0 ⊕1 ⊕0 ⊕1 = 1

h2 = m1 ⊕m3 ⊕m4 ⊕m6 ⊕m7 = 1 ⊕1 ⊕1 ⊕1 ⊕1 = 1

h3 = m2 ⊕m3 ⊕m4 ⊕m8 = 0 ⊕1 ⊕1 ⊕0 = 0

h4 = m5 ⊕m6 ⊕m7 ⊕m8 = 0 ⊕1 ⊕1 ⊕0 = 0

Si ricava, quindi, il seguente codice Hamming:

MS = **11**1**0**011**0**0110

dove in neretto sono individuati i bit di controllo.

L’entità che riceve il frame verifica la sua bontà **mediante un controllo che si basa sull’XOR tra tutti i bit controllati con il proprio bit controllore determinando le somme Si**, così come mostrato negli esempi che seguono. Dal valore delle somme Si dipende la correttezza del messaggio; i casi possibili sono due; ovvero:

1. messaggio corretto 🡪 tutti gli Si sono uguali a 0;

2. messaggio errato 🡪 esiste almeno un Si uguale a 1. In tal caso i valori delle quattro somme S4S3S2S1 indicano la posizione del bit errato.

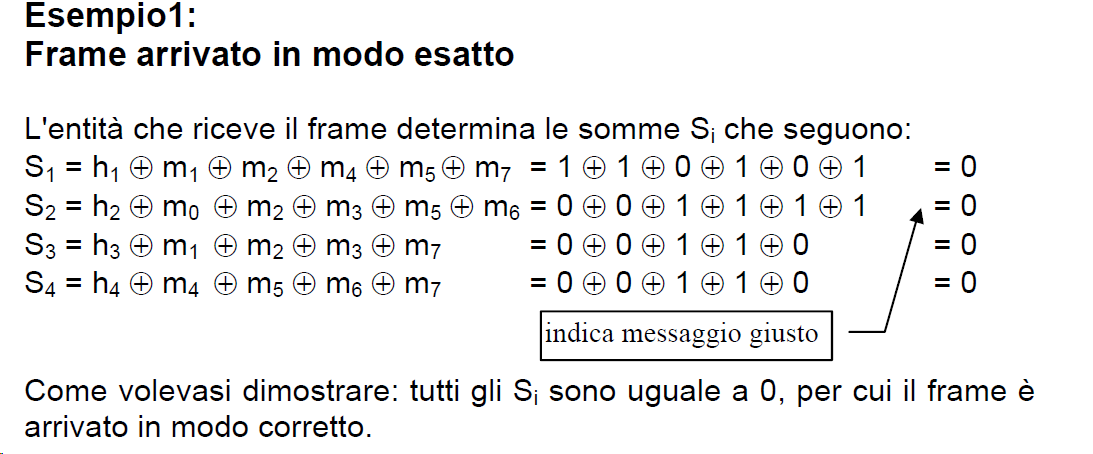
Allo scopo vediamo tre esempi in cui si verificano i seguenti eventi:

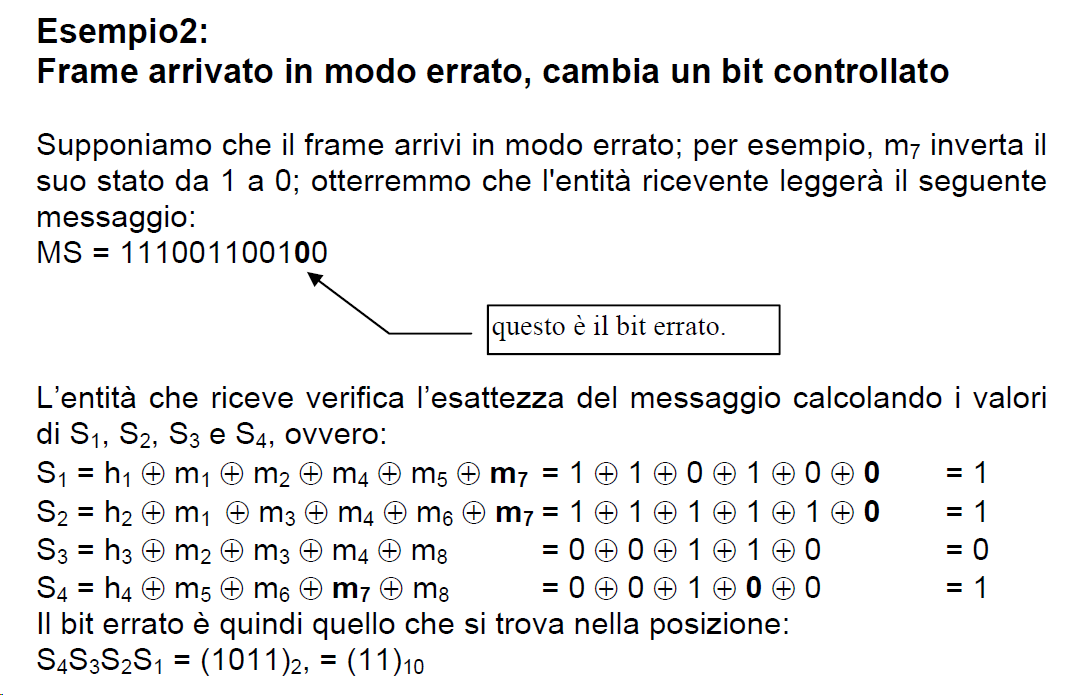
1. frame arrivato in modo esatto;

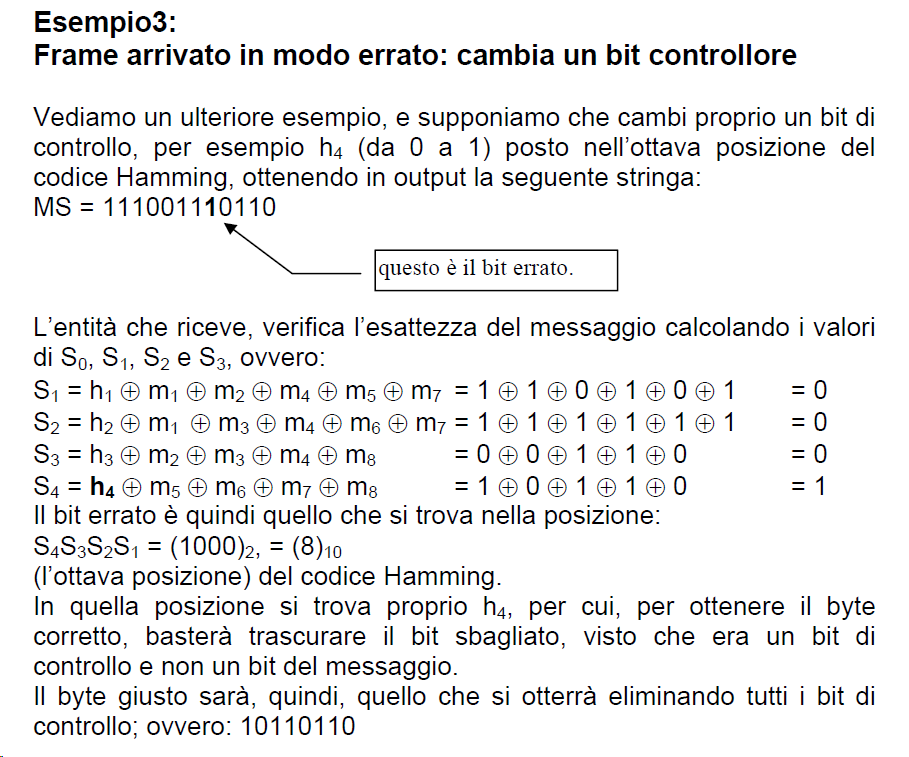
2. frame arrivato in modo errato perché è cambiato un bit del messaggio

(bit controllato);

3: frame arrivato in modo errato perché è cambiato un bit controllore.







I codici correttori di errore, ad eccezione delle trasmissioni simplex dove non è possibile inviare al

mittente una richiesta di ritrasmissione, sono usati molto raramente.

E’ più efficiente limitarsi a rilevare gli errori, e ritrasmettere saltuariamente i dati errati, piuttosto

che impiegare un codice (più dispendioso in termini di ridondanza) per la correzione degli

errori.

## Codici a rilevazione d’errore