1 IL SEGNALE

1.1 Il segnale analogico

Tutti i dispositivi per comunicare si scambiano dei segnali. Attraverso l'aria i tablet e gli smartphone, attraverso i cavi i computer desktop. I segnali portano anche le trasmissioni televisive e quelle radiofoniche, terrestri e satellitari. Con i segnali viaggiano tutti i dati necessari a trasferire suoni, immagini, video, file, pagine web, ecc. che sono rappresentati sotto forma di lunghe sequenze di bit.

Esattamente come avviene all'interno di un computer, anche per le trasmissioni a distanza occorre quindi trasferire da un punto all'altro dei bit.

L'informazione può essere trasmessa a distanza variando opportunamente una qualche caratteristica del fenomeno fisico utilizzato per la trasmissione. Associando a questa variazione uno dei due valori del bit e alla mancanza di variazione (o a una diversa variazione) l'altro valore del bit, si realizza la trasmissione dei dati.
Tale variazione si propaga, con una certa velocità, lungo il mezzo di trasmissione e dopo un certo tempo arriva all'altra estremità del mezzo, dove può essere rilevata. Per esempio, se il mezzo è un cavo elettrico, si può variare la tensione applicata a un'estremità. Tale variazione di tensione verrà successivamente rilevata all'altra estremità.

Qualunque sia il mezzo trasmissivo e il fenomeno fisico utilizzato, i segnali trasmessi possono essere di due tipi: **analogici** e **digitali**.

I **segnali analogici** possono assumere un qualsiasi valore all'interno di un determinato intervallo senza soluzione di continuità. In particolare un segnale analogico **periodico**, cioè che assume valori che si ripetono ciclicamente in modo regolare nel tempo, è particolarmente adatto a trasportare i bit.

I segnali periodici più utilizzati sono quelli **sinusoidali**, che consentono di comporre e descrivere qualsiasi altro segnale periodico (teorema di Fourier).
Ogni segnale sinusoidale fa riferimento a una grandezza che varia nel tempo e che viene scelta per descrivere il segnale, per esempio una differenza di potenziale, un'intensità di corrente o un'intensità luminosa.
Per descrivere un segnale sinusoidale periodico si utilizzano tre parametri:

- **ampiezza**: la distanza tra il valore medio e quello massimo della grandezza scelta;
- **frequenza**: il numero di volte in cui si ripete il segnale in un secondo (viene misurata in hertz);
- **fase**: intervallo di tempo, espresso in gradi, tra l'inizio di un segnale sinusoidale e un tempo prefissato preso come riferimento. Ne deriva che lo **sfasamento** tra due segnali è l'intervallo di tempo, sempre espresso in gradi, che intercorre tra due segnali sinusoidali con la stessa frequenza.

Nella FIGURA 1 è mostrato un grafico con un segnale analogico periodico (l'asse orizzontale indica il tempo, quello verticale l'ampiezza dell'onda sinusoidale), in cui l'onda disegnata ha un'ampiezza di 5 volt a 0,25 secondi, di 0 volt a 0,5 secondi e di –5 volt a 0,75 secondi (come grandezza è stata usata la tensione da +5 V a –5 V).

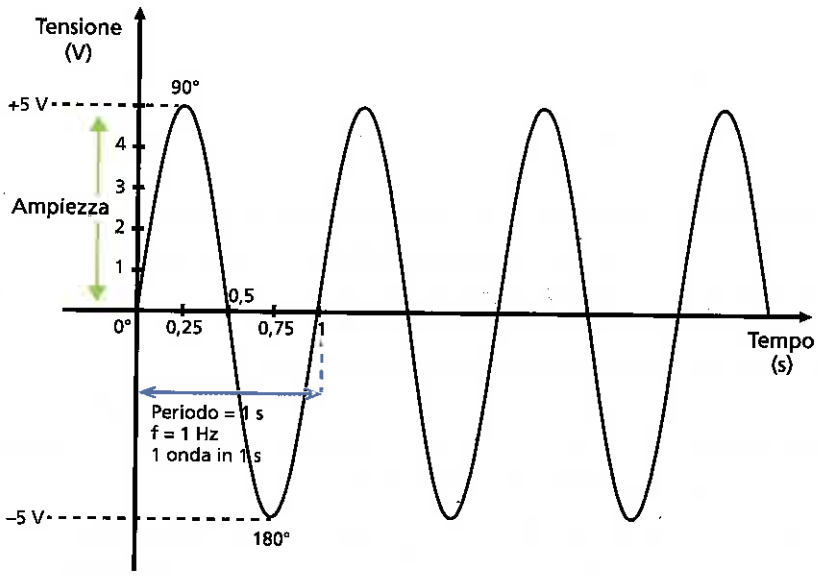


FIGURA 1 Segnale analogico sinusoidale periodico

La frequenza dell'onda è di 1 Hz, ossia di un ciclo ogni secondo. Infatti il ciclo inizia dal punto 0, prosegue fino a raggiungere l'ampiezza massima (+5), scende poi fino al punto minimo (–5) e quindi ritorna al punto iniziale; il tutto avviene nel tempo di un secondo.

Nella FIGURA 2 sono mostrate due diverse forme d'onda che rappresentano due segnali analogici e hanno lo stesso periodo, la stessa ampiezza, la stessa frequenza, ma fase diversa. Infatti un ciclo completo equivale a 360°, dove 0° rappresenta il punto di partenza, 90° il picco positivo, 180° il punto in cui interseca l'asse orizzontale, 270° il picco negativo e 360° la conclusione del ciclo. I due segnali analogici rappresentati in figura hanno una differenza di fase di 90°. Il calcolo dello sfasamento tra due segnali con la stessa frequenza si ottiene tenendo conto che alla durata di un periodo, indicato con T, corrispondono 360°, quindi si misura l'intervallo I e il periodo T sull'asse del tempo e successivamente si imposta la proporzione:

$$I : T = \theta : 360°$$

dove θ indica lo sfasamento.

Risulta quindi che $\theta = (360° \cdot I)/T$ (espresso in gradi).

Nell'esempio si ha quindi: $\theta = (360° \cdot 0{,}25)/1 = 90°$.

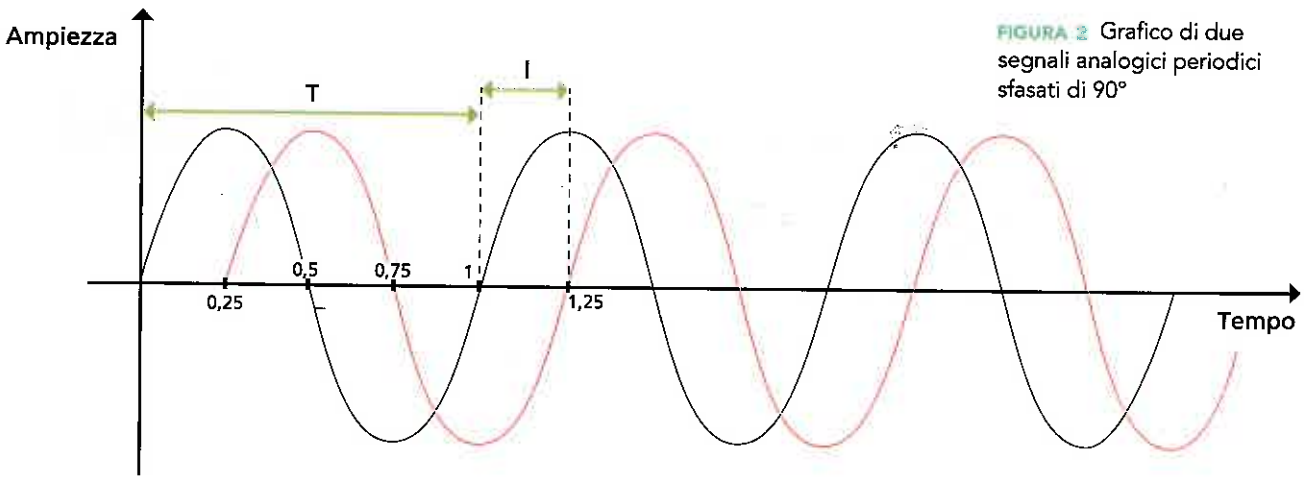


FIGURA 2 Grafico di due segnali analogici periodici sfasati di 90°

1.2 Il segnale digitale

I **segnali digitali** hanno due caratteristiche che li distinguono dai segnali analogici:

- possono assumere solo un numero limitato di valori discreti (due nel caso di segnali binari);
- la transizione da un valore all'altro avviene in modo quasi istantaneo.

Per queste caratteristiche un segnale digitale è rappresentato con un'onda rettangolare, che nel caso di segnali digitali binari è costituita da due valori: uno alto (*high*) che rappresenta un bit 1 e uno basso (*low*) che rappresenta un bit 0.
Nella FIGURA 3 possiamo vedere un segnale digitale binario costituito da 8 bit (un byte).

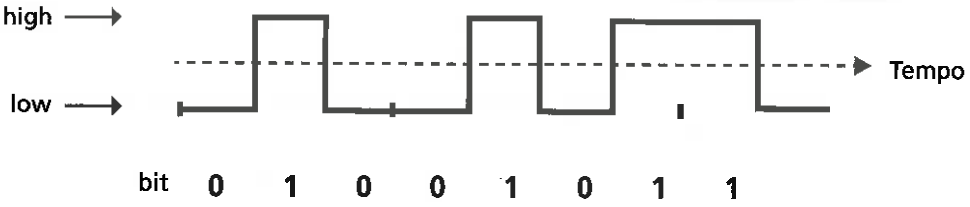


FIGURA 3 Segnale digitale binario

I segnali digitali, e in particolare quelli binari, godono di alcuni importanti pregi nei confronti di quelli analogici.

MAGGIOR ROBUSTEZZA RISPETTO AL RUMORE

Rumore e disturbi, per quanto possano essere minimi, sono sempre presenti, sia nel trasmettitore, sia nel ricevitore, ma sono soprattutto presenti nel mezzo di comunicazione, sia esso un cavo o l'aria. I segnali digitali per loro natura sono meno soggetti ai disturbi rispetto ai segnali analogici. I segnali analogici sono costituiti da funzioni continue che possono assumere infiniti valori: il rumore inevitabilmente si sovrappone al segnale trasmesso e lo modifica rendendo difficile risalire al segnale originario. I segnali digitali, invece, presentano solamente un numero finito di valori separati da un "salto" tra uno e l'altro. Tale salto è determinato dal superamento di una soglia. Se il rumore non ha ampiezza (e potenza) tale da determinare il superamento della soglia e quindi il salto tra due valori, allora il rumore non riesce ad alterare il segnale. Il destinatario, ricevendo il valore non ammissibile, provvederà a convertirlo nel valore accettabile più vicino, annullando l'effetto del disturbo (**autorigenerazione**). Naturalmente se il rumore fosse tale da far superare la soglia, il destinatario sarebbe indotto in errore e il segnale verrebbe convertito in un valore sbagliato in assenza di tecniche di correzione. Una delle ragioni che fanno preferire l'utilizzo dei segnali digitali rispetto a quelli analogici è proprio l'autorigenerazione: i segnali digitali contengono errori che, al di sotto di una certa soglia, si autocorreggono. In un segnale digitale binario solo parzialmente modificato da un disturbo (FIGURA 4A), proprio la limitazione ai soli valori "0" e "1" rende i bit ancora riconoscibili (FIGURA 4B).

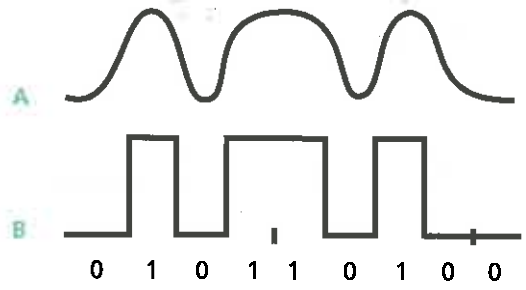


FIGURA 4 Segnale digitale binario autorigenerato

MAGGIOR INTEGRAZIONE DEI SISTEMI DI TRASMISSIONE

La conversione in digitale di tutte le trasmissioni (audio, video, dati, testo, ecc.) rende ogni invio simile a tutti gli altri: un flusso di bit con la sola aggiunta di intestazioni (bit di *header*) che specificano il tipo di informazioni trasmesse in ogni blocco. Qualunque apparato dotato di connessione di rete è in grado di inviare e ricevere qualsiasi tipo di trasmissione digitale. Anche il mezzo fisico (cavo, fibra, aria) e la tecnologia (elettrica, ottica, wireless) diventano irrilevanti. I flussi di bit, raggruppati in byte e poi in pacchetti (*packet*) aventi regole precise, uniformano le fasi di invio, trasferimento e ricezione.

MAGGIOR ADATTAMENTO A ESSERE ESEGUITI E MEMORIZZATI

Il linguaggio dei segnali digitali è lo stesso dei microprocessori: un linguaggio binario. Con le opportune interfacce, le sequenze di bit trasmesse possono facilmente essere lette ed eventualmente eseguite o memorizzate.
I dispositivi di memoria sono in grado di conservare grosse quantità di dati con l'utilizzo di tecniche digitali. La presenza di due soli stati fisici nei supporti di memoria, associati rispettivamente a 0 e 1, rende la tecnologia dei circuiti integrati in grado di leggere e scrivere sempre più velocemente.

MAGGIORE ADATTABILITÀ A TECNICHE DI ELABORAZIONE DEL SEGNALE

Mentre l'elaborazione dei segnali analogici è generalmente limitata alle operazioni di amplificazione, di modulazione e di filtraggio, l'elaborazione dei segnali digitali può consentire operazioni complesse:

- **rivelazione e correzione degli errori**: nei sistemi digitali si possono realizzare circuiti e algoritmi per la rivelazione e la correzione degli errori in trasmissione, come vedremo nella Lezione 4 di questa unità;
- **crittografia**: con un sistema digitale l'informazione è codificata ed è quindi possibile adottare forme di crittografia per rendere incomprensibili, tranne che al destinatario, le informazioni trasmesse;
- **incapsulamento**: protezione dei dati in trasmissione con l'aggiunta di ulteriori header;
- **privacy**: modifica degli indirizzi privati dei mittenti e dei destinatari;
- **compressione**: riduzione della quantità di dati digitali da trasmettere, comprimendo opportunamente il segnale ed evitando di ripetere l'invio delle informazioni che si ripetono uguali (ad esempio le immagini in sequenza hanno spesso sfondi che restano a lungo invariati).

FISSA LE CONOSCENZE

- Descrivi le differenze tra segnale analogico e segnale digitale (massimo 5 righe).
- Quali sono i tre parametri che descrivono un segnale analogico sinusoidale?
- Che cosa si intende per sfasamento tra due segnali analogici?
- Quali valori assume un segnale digitale binario?
- Che cosa si intende con autorigenerazione di un segnale digitale?

2 LE MODULAZIONI DIGITALI

2.1 Modulare e demodulare

La trasmissione dati si basa sulla trasmissione digitale, però ci possono essere casi in cui il tipo di connessione utilizzata è in grado di trasmettere solo segnali analogici (per esempio le linee telefoniche sono state progettate per trasportare segnali analogici). Se ci si connette a una rete geografica, come Internet, tramite una linea telefonica, i dati che escono dal computer (digitali) devono essere convertiti in segnali analogici prima di essere inviati sulla linea telefonica. In caso di ricezione dati avviene il processo inverso: i segnali analogici che giungono dalla linea devono essere convertiti nella forma originaria digitale. Questo processo è chiamato **modulazione/demodulazione** ed è realizzato tramite un apparato denominato **modem**.

Nella **modulazione**, un'onda, chiamata **segnale portante** (carrier signal), è combinata con un altro segnale, chiamato **segnale modulante** (modulating signal), per produrre un unico segnale che trasporta l'informazione da un sistema a un altro. Quando il segnale modulante è combinato al segnale portante, esso modifica un parametro del segnale portante, per esempio la frequenza o l'ampiezza o la fase. Il risultato è un nuovo segnale che viene inviato sul mezzo trasmissivo e quando arriva a destinazione l'apparato ricevente separa il segnale modulante da quello portante (**demodulazione**), ricostruendo in tal modo il segnale originario.

Le modulazioni si basano sulla trasformazione della sequenza di bit in un segnale digitale modulante che, opportunamente combinato con una portante analogica sinusoidale, origina il segnale analogico da trasmettere.
Le modulazioni digitali fondamentali sono:

- **ASK** (Amplitude Shift Keying): modulazioni a cambiamento di ampiezza;
- **FSK** (Frequency Shift Keying): modulazioni a cambiamento di frequenza;
- **PSK** (Phase Shift Keying) e **DPSK** (Differential PSK): modulazioni a cambiamento di fase;
- **QAM** (Quadrature Amplitude Modulation): modulazioni di più bit alla volta.

2.2 ASK (Amplitude Shift Keying)

Nella ASK il segnale modulante decide tra due diverse ampiezze di una portante sinusoidale, che per semplicità di rappresentazione grafica supporremo a frequenza 1 Hz (un solo segnale in un periodo).
La più usata, mostrata in FIGURA 5, è la **OOK** (On-Off Keying), che associa all'1 la presenza della portante (con ampiezza invariata) e allo 0 l'assenza della portante (ampiezza zero).

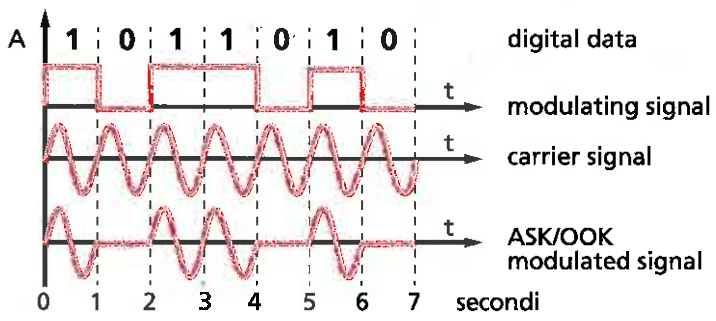


FIGURA 5 Modulazione OOK di tipo ASK

È possibile realizzare modem che lavorano su due diversi valori di ampiezza, sempre da associare all'1 e allo 0.
La ASK è stata la prima modulazione a essere usata (telescriventi e ponti radio) ed è la più semplice da realizzare. È però caduta in disuso per l'estrema sensibilità al rumore.

2.3 FSK (Frequency Shift Keying)

Nella FSK il segnale modulante decide tra due portanti sinusoidali a frequenze diverse (FIGURA 6). Alla frequenza di una portante è associato l'1, alla frequenza dell'altra lo 0. Nella figura si sono usate la frequenza 1 Hz per i bit a 1 e la frequenza 2 Hz (due cicli in un periodo) per i bit a 0. La FSK è stata utilizzata dai primi modem per Internet (voce e dati) e per le trasmissioni tra cellulari GSM.

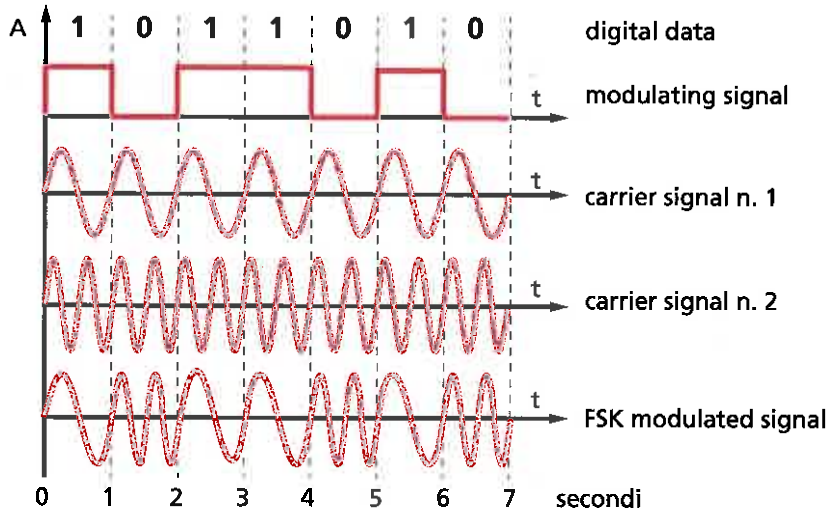


FIGURA 6 Modulazione di tipo FSK

2.4 PSK (Phase Shift Keying)

Nella PSK il segnale modulante decide tra due diverse fasi di una portante sinusoidale (FIGURA 7).
A una fase della portante è associato l'1, all'altra fase lo 0.
La PSK utilizza le fasi 0° (nessuno sfasamento della portante) e 180° (sfasamento di 180° della portante) rispettivamente per l'1 e lo 0 ed è la più attuale tra le modulazioni fondamentali.

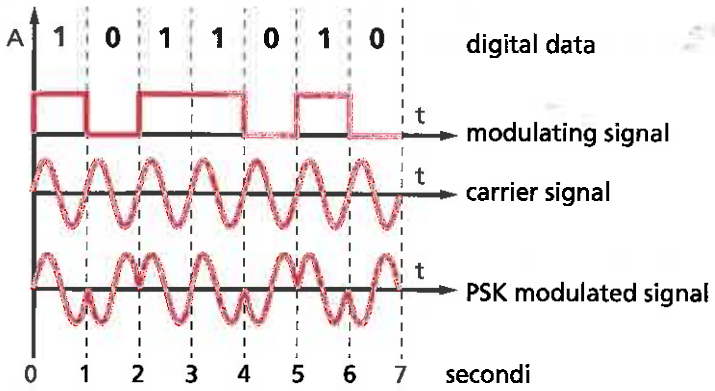


FIGURA 7 Modulazione di tipo PSK

Esiste anche una variante della PSK, chiamata **DPSK** (Differential PSK), che, invece di associare un'onda ben precisa e sempre la stessa a ogni bit, segue una regola basata sullo **sfasamento rispetto al bit precedente**: al bit 0 non corrisponde alcun cambio di fase rispetto al bit precedente, al bit 1 corrisponde un cambio di fase di 180° rispetto al bit precedente. Serve quindi un **segnale di riferimento** prima del primo bit da trasmettere, poi ogni bit fa da riferimento per il bit successivo. Mittente e destinatario devono accordarsi sul segnale di riferimento da usare. Quindi con la DPSK, a differenza di tutte le altre tecniche viste, uno stesso bit può essere rappresentato a volte da un'onda e a volte da un'onda opposta: dipende dal bit che lo precede.
La FIGURA 8 mostra il confronto tra le due modulazioni. In questa figura ogni bit è rappresentato da un segnale con una frequenza di 2 Hz..

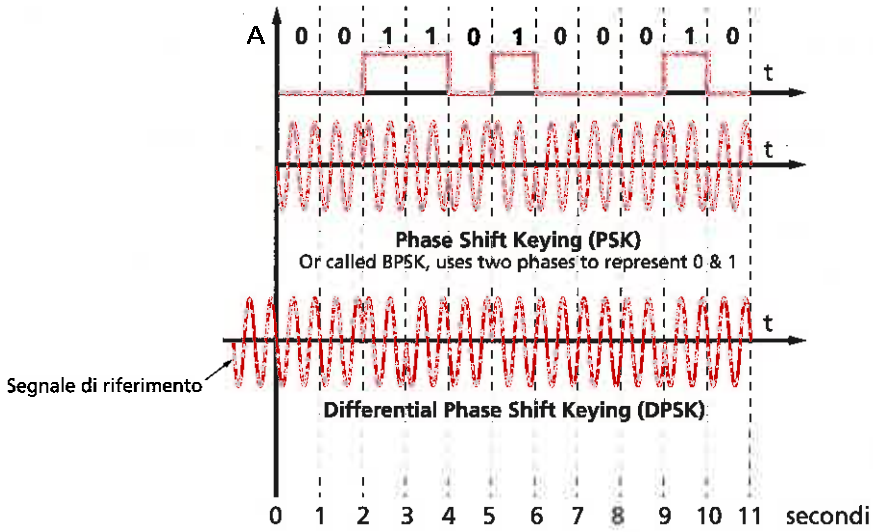


FIGURA 8 Modulazione di tipo PSK e DPSK a confronto

Le tre modulazioni digitali fondamentali appena viste hanno una caratteristica comune: modulano un bit per volta, cioè associano un'onda o una regola (di sfasamento) a un singolo bit.
In altre parole, ogni periodo del segnale sinusoidale trasmesso porta un bit di informazione.
È possibile fare in modo che un'onda porti più di un bit d'informazione attraverso le modulazioni QAM.

2.5 QAM (Quadrature Amplitude Modulation)

Le prime modulazioni di tipo QAM trasportavano 2 bit per volta utilizzando 4 fasi e creando quindi un'associazione del tipo PSK, come per esempio:

$$00 \rightarrow 0°$$
$$01 \rightarrow 90°$$
$$10 \rightarrow 180°$$
$$11 \rightarrow 270°$$

nota come 4-QAM dibit (FIGURA 9). Il modem realizza uno **sfasamento** tra un segnale e l'altro pari ai gradi indicati. Il procedimento è dunque simile alla DPSK, ma utilizzando 4 fasi anziché 2 è possibile trasmettere una coppia di bit anziché un bit solo. Quindi, anche in questo caso, una coppia di bit non è sempre espressa dalla stessa onda e serve un segnale di riferimento per la prima coppia di bit.

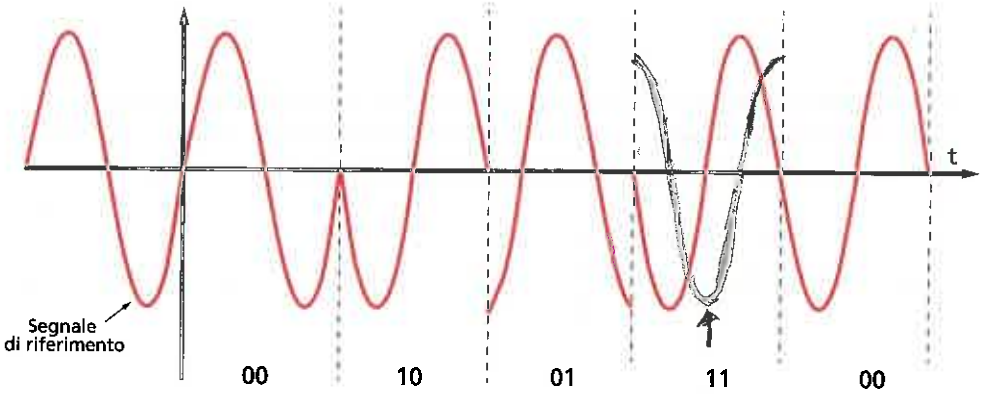


FIGURA 9 Modulazione di tipo 4-QAM dibit

Con 8 fasi (usando per esempio i valori multipli di 45°) si possono trasportare 3 bit per modulazione (8-QAM tribit), con 16 fasi 4 bit (16-QAM quadribit) e così via.
Vale la regola:

$$\text{numero fasi} = 2^{(\text{numero bit})} \quad \text{oppure} \quad \text{numero bit} = \log_2(\text{numero fasi})$$

Poiché c'è la necessità di trasmettere ancora più bit per volta senza correre rischi di errore in demodulazione dovuti a sfasamenti troppo ravvicinati, si è pensato di aggiungere contemporaneamente una variazione di ampiezza (PSK + ASK).

> Le modulazioni QAM sono le uniche che modulano più di un bit per volta, associandoli a più valori di una stessa grandezza (di solito la fase) oppure agendo su due grandezze (di solito ampiezza e fase), mentre ogni altra modulazione modifica una sola delle grandezze che caratterizzano un segnale analogico (o ampiezza o frequenza o fase) e lascia inalterate le altre due.

Un esempio è dato dalla 16-QAM quadribit, realizzata con 12 fasi e 4 ampiezze (±A e ±3A), che dà origine alla *costellazione* mostrata in FIGURA 10.

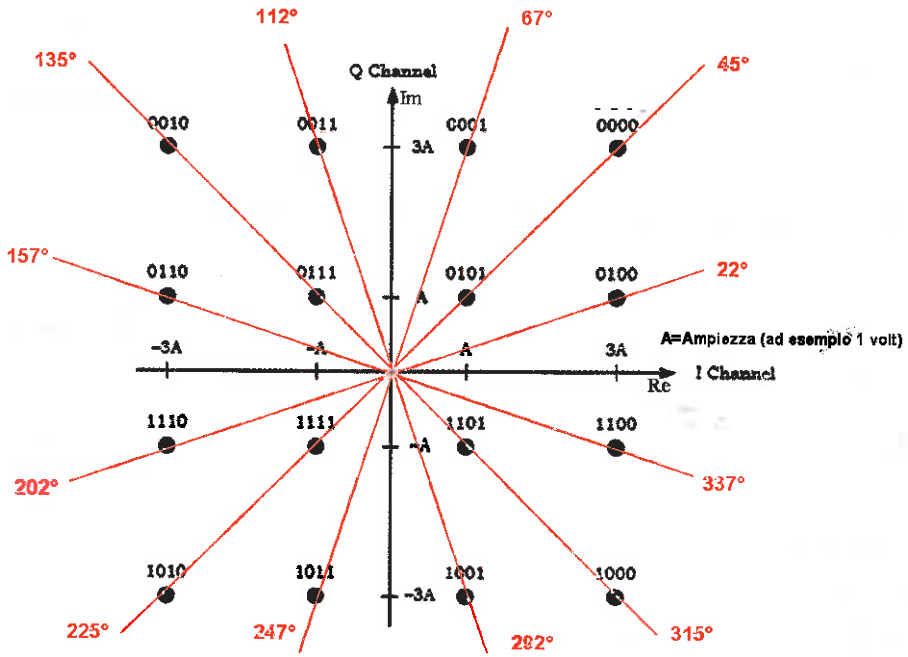


FIGURA 10 Costellazione della modulazione di tipo 16-QAM quadribit

I valori di ampiezza sono rappresentati in componente reale e immaginaria rispettivamente sugli assi I Channel e Q Channel.
Per esempio, la stringa di dati digitale 1001 è associata a un'onda sfasata di 292° e con ampiezza A in componente reale e –3A in componente immaginaria.
Le modulazioni QAM, che consentono la trasmissione di più bit con un'unica modulazione, introducono il concetto di #baud rate.
Nella FIGURA 11 vediamo un esempio di 8-QAM tribit (4 fasi e due ampiezze), che in un secondo trasmette 24 bit (bit rate = 24 bps) con un baud rate di 8 baud/s.

#techwords

Il **baud rate** è il numero di segnali diversi trasmessi in un secondo.

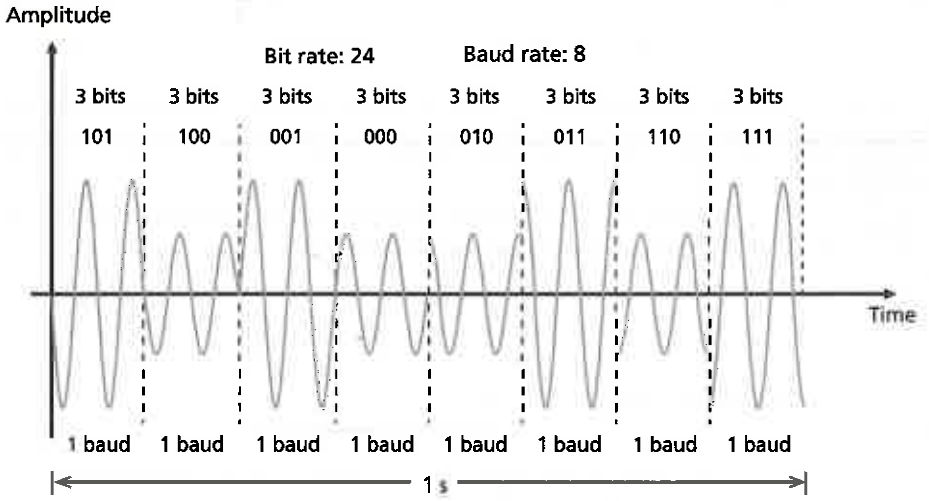


FIGURA 11 Bit rate e baud rate

IN ENGLISH PLEASE

Bit rate is the number of bits per second. Baud rate is the number of signal units per second. Baud rate is less than or equal to the bit rate.

Example 1

An analog signal carries 4 bits in each signal unit. If 1000 signal units are sent per second, find the baud rate and the bit rate.

Solution

Baud rate = 1000 baud/s
Bit rate = 1000 x 4 = 4000 bps

Example 2

The bit rate of signal is 3000. If each signal unit carries 6 bits, what is the baud rate?

Solution

Boud rate = 3000/6 = 500 baud/s

FISSA LE CONOSCENZE

- Che cosa si intende con modulazione di un segnale?
- Descrivi, anche aiutandoti con un grafico, come funziona la FSK (*Frequency Shift Keying*).
- Descrivi, anche aiutandoti con un grafico, come funziona la DPSK (*Differential Phase Shift Keying*).
- Che differenza c'è tra bit rate e baud rate?

3 IL CANALE DI COMUNICAZIONE

3.1 Il multiplexing del canale

Quando un segnale deve essere trasmesso, viene inviato attraverso un **#canale**, cioè un mezzo fisico di trasmissione.

#techwords

Un **canale** (channel) è un percorso definito che mette in comunicazione due nodi.

I canali possono essere logici o fisici: sono **logici** quando si realizzano più percorsi distinti utilizzando lo stesso mezzo fisico (per esempio lo stesso cavo di rame), sono **fisici** quando si realizzano più percorsi distinti utilizzando mezzi fisici diversi.

La tecnica che consente di separare un mezzo fisico in più canali logici viene detta **multiplexing** e permette di far viaggiare più segnali simultaneamente su uno stesso mezzo fisico.
Esistono diverse tecniche per effettuare il multiplexing dei segnali: TDM (Time Division Multiplexing), FDM (Frequency Division Multiplexing), WDM (Wavelength Division Multiplexing).

TDM (TIME DIVISION MULTIPLEXING)

Questa tecnica prevede di suddividere ogni frame in intervalli (slot) e assegnare ogni slot a un dispositivo di input (FIGURA 12). In questo modo ogni frame inviato contiene una parte di dati di ciascun mittente. Il risultato è quello di portare avanti più trasmissioni simultaneamente, distribuendo il canale fisico, anziché concederlo a turno a un solo mittente per volta.

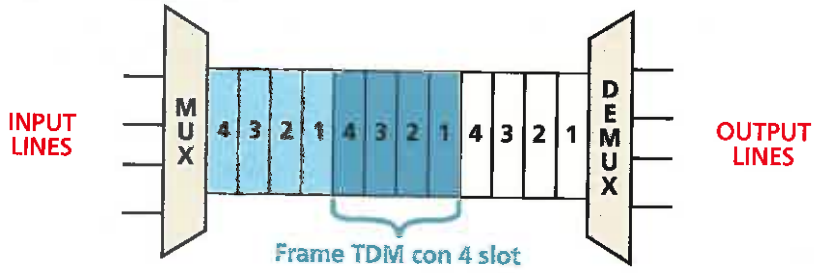


FIGURA 12 Tecnica TDM

Il dispositivo di multiplexing (MUX) è dotato di un buffer, in cui prepara il frame prima di inviarlo. Qualora non tutte le linee di input avessero dati da trasmettere, il MUX può comunque trasmettere il frame con bit riempitivi (padding) o riempirlo, assegnando più slot a uno stesso mittente. Il dispositivo di demultiplexing (DEMUX) mantiene su ciascuna linea di output lo slot relativo al destinatario corrispondente.

FDM (FREQUENCY DIVISION MULTIPLEXING)

Questa tecnica prevede invece di suddividere il canale in **sottocanali**, uno per ciascun mittente (FIGURA 13).

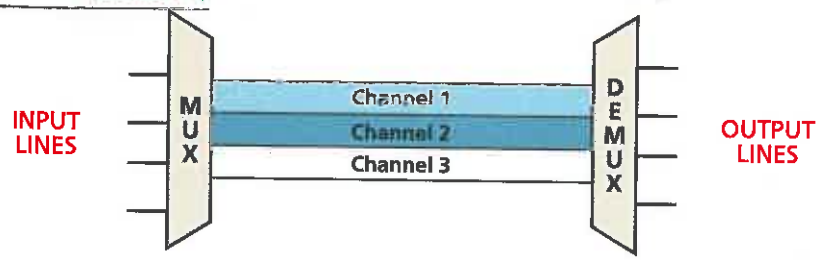


FIGURA 13 Tecnica FDM

In pratica, ogni sottocanale lavora a frequenze diverse dagli altri (FIGURA 14), evitando interferenze reciproche.

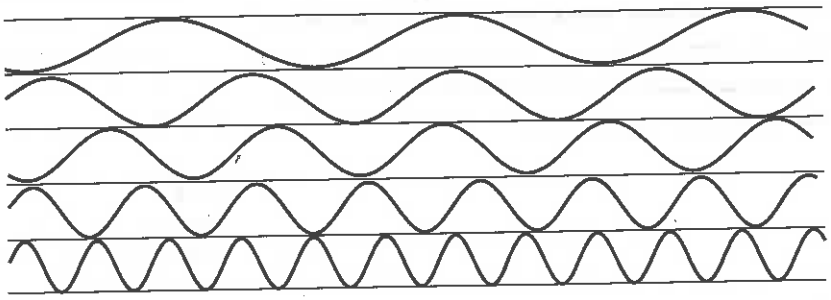


FIGURA 14 Sottocanali a frequenze diverse

WDM (WAVELENGTH DIVISION MULTIPLEXING)

È una tecnica implementata nelle attuali reti in fibra ottica, che permette di usare in simultanea su una singola fibra tra gli 80 e i 160 canali logici, trasmessi su diverse lunghezze d'onda (FIGURA 15).

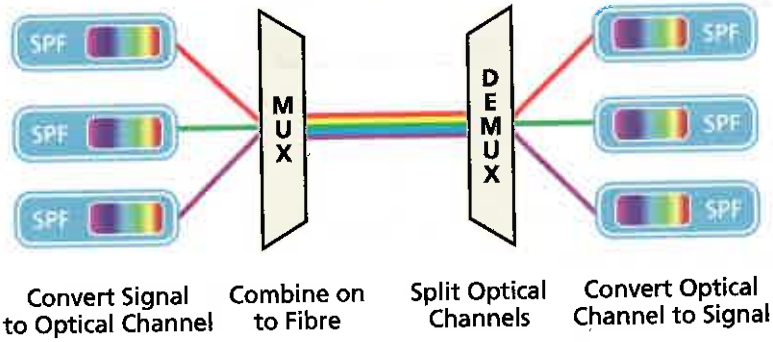


FIGURA 15 Tecnica WDM

Il multiplexing può essere di tipo denso, quando la differenza tra le lunghezze d'onda (FIGURA 16) di canali adiacenti è inferiore a 1 nm (nanometro).

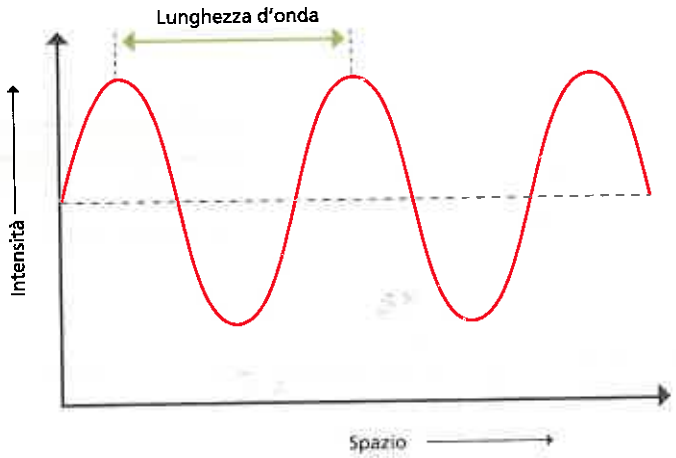


FIGURA 16 Lunghezza d'onda

La lunghezza d'onda di un'onda sinusoidale è la distanza, espressa in nanometri, percorsa dall'onda durante un ciclo completo di oscillazione (periodo). Il nanometro (nm) è un sottomultiplo del metro ed equivale a un miliardesimo di metro: 1 nm = 10^{-9} m
In un'onda elettromagnetica tra lunghezza d'onda e frequenza esiste un rapporto di proporzionalità inversa.

IN ENGLISH PLEASE

The **WDM** unit consists of a Signal Processing Facility (**SPF**), a multiplexer (**MUX**) which is responsible for joining the signals at the transmitting end, and a demultiplexer (**DEMUX**) which is responsible for splitting the signals at the receiving end.

CWDM and DWDM

There are two types of WDM: **Coarse** and **Dense** Wavelength Division Multiplexing (CWDM and DWDM).

CWDM uses a wide spectrum and accommodates eight channels. This wide spacing of channels allows for the use of moderately priced optics, but limits capacity. CWDM is typically used for lower-cost, lower-capacity, shorter-distance applications where cost is the paramount decision criteria.

DWDM systems pack 16 or more channels into a narrow spectrum window very near the 1550 nm local attenuation minimum. Decreasing channel spacing requires the use of more precise and costly optics, but allows for significantly more scalability. Typical DWDM systems provide 1-44 channels of capacity, with some new systems, offering up to 80-160 channels. DWDM is typically used where high capacity is needed over a limited fiber resource or where it is cost prohibitive to deploy more fiber.

Product Overview

The Cisco EWDM product line provides the ability to overlay up to 8 DWDM wavelengths with the 8 CWDM channels (1470, 1490, 1510, 1530, 1550, 1570, 1590, and 1610 nm). The principle is very simple, yet it is a unique approach in that the 8 DWDM wavelengths are inserted in between CWDM channels. EWDM allows 5 DWDM channels to be multiplexed between the 1530-nm and 1550-nm CWDM wavelengths and 3 DWDM channels between the 1550-nm and 1570-nm CWDM wavelengths. A total of 8 CWDM plus 8 DWDM wavelengths can be supported on the same fiber infrastructure (see FIGURES 17, 18, 19).

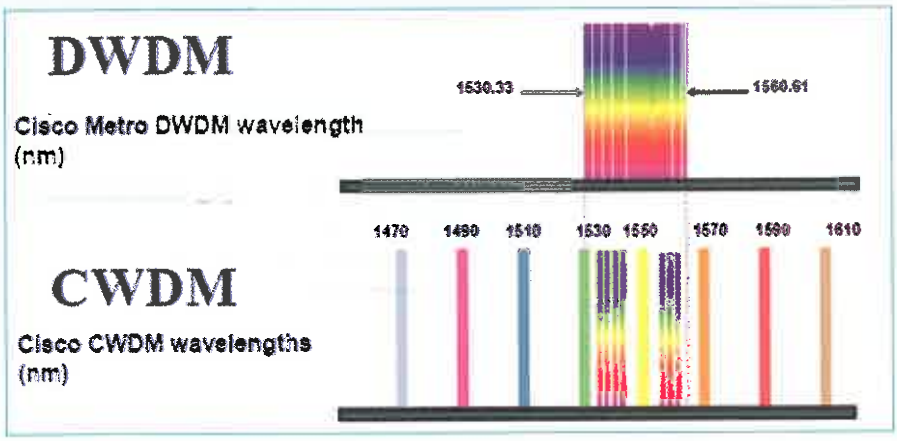


FIGURA 17 Cisco EWDM Concept

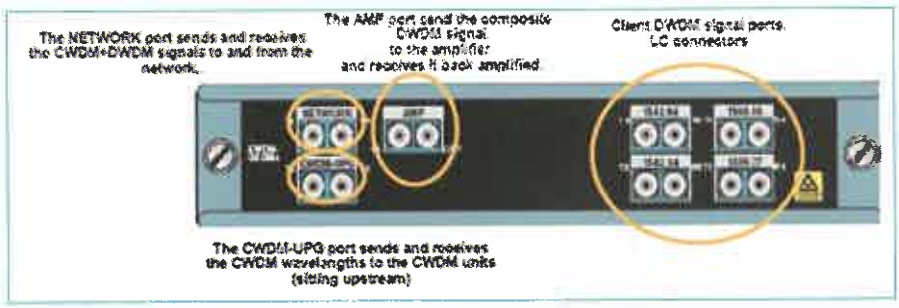


FIGURA 18 Cisco EWDM Passive Unit Front Panel Layout

The Cisco® enhanced wavelength-division multiplexing (EWDM) product line allows users to scale the speed and capacity of the services offered in a coarse wavelength-division multiplexing (CWDM) network by offering the ability to insert up to 8 dense wavelength-division multiplexing (DWDM) wavelengths to the existing 8-wavelength CWDM channel plan.

FIGURA 19 LC optical connector

3.2 La codifica di linea

I dati da trasmettere sul canale devono essere prima trasformati in un segnale elettrico con la **codifica di linea** (in ricezione si utilizza la **decodifica di linea**) che serve ad adattare il segnale fisico digitale al particolare mezzo trasmissivo utilizzato. Inoltre, la codifica di linea deve permettere di mantenere il **sincronismo** tra trasmettitore e ricevitore.

Il segnale di sincronismo è il segnale di clock che sincronizza le schede di rete delle macchine, quindi una caratteristica importante della codifica di linea è quella di mettere insieme i dati con il segnale di sincronismo, cosa che permette al ricevitore di effettuare correttamente la decodifica del segnale ricevuto.

La trasmissione dei dati in forma digitale prevede che ai valori 1 e 0 dei bit da inviare si associno determinati valori del fenomeno fisico che è stato scelto per la trasmissione (per esempio la differenza di potenziale). La semplice scelta di associare due differenti valori fisici (uno per lo 0 e uno per l'1) non è quella ottimale, in quanto comporta problemi di sincronismo tra trasmettitore e ricevitore.

Le tre principali tecniche di codifica sono: **NRZ** (Not Return to Zero), molto semplice, usata nei computer e nelle centrali numeriche, **RZ** (Return to Zero), usata nelle centraline telefoniche, **Manchester** e le sue varianti, usate soprattutto nelle reti dati locali. La FIGURA 20 riassume tutte le tecniche di seguito descritte.

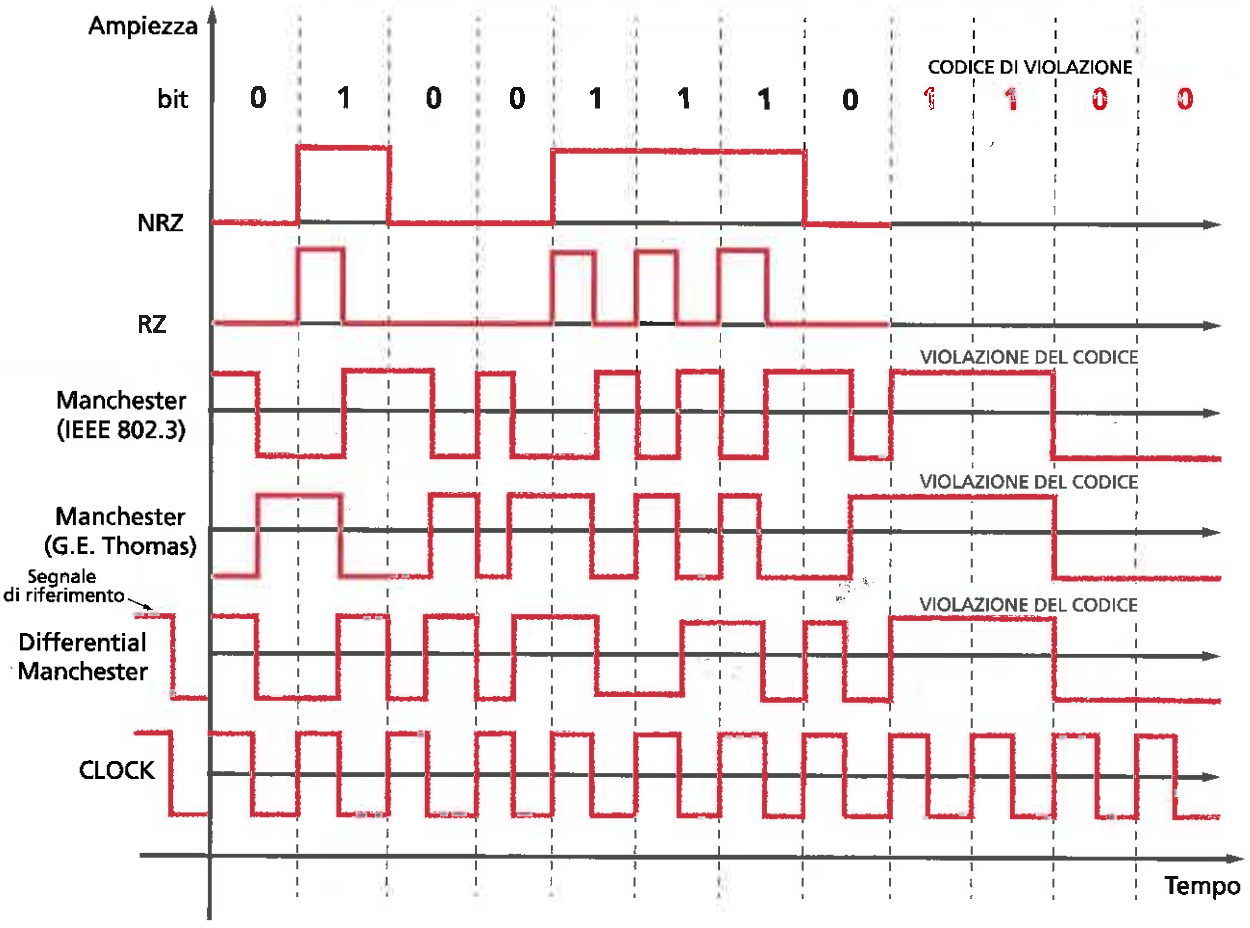


FIGURA 20 Le differenti codifiche di linea per una stessa sequenza di bit

NRZ (NOT RETURN TO ZERO)

È la codifica più semplice e associa un valore alto al bit 1 e un valore basso al bit 0. Si tratta di un metodo che non richiede circuiti complicati perché i dati che entrano come 1 o 0 vengono passati direttamente all'uscita senza modifiche. Il problema principale di questa codifica è la difficoltà a mantenere il sincronismo a fronte di una lunga sequenza di bit uguali (tutti 1 o tutti 0) che porta il segnale ad avere lo stesso valore per un lungo intervallo di tempo: un minimo disallineamento nel clock del ricevitore comporterà un'interpretazione errata della sequenza di bit.

RZ (RETURN TO ZERO)

È simile a NRZ, con la differenza di portare il segnale a zero a ogni semiperiodo. Il bit 1 è quindi rappresentato da un valore alto per metà del periodo di clock e poi da un valore basso per la restante metà. Questa codifica risolve il problema di lunghe sequenze di bit 1 (valore alto) ma non di quelle di bit 0 (valore basso).

MANCHESTER

In questa codifica, definita all'Università di Manchester da cui prende il nome, il segnale di clock del trasmettitore e il segnale dei dati vengono combinati per garantire una transizione per ogni bit. Esistono due opposte convenzioni, entrambe con numerosi sostenitori, su come rappresentare il bit 1 e il bit 0:

- la prima è specificata nello standard Ethernet (IEEE 802.3) da cui il nome Manchester 802.3 e afferma: il bit 1 è rappresentato da una transizione basso-alto al semiperiodo e il bit 0 da una transizione alto-basso al semiperiodo;
- la seconda è quella proposta da G.E. Thomas, che specifica l'opposto: il bit 1 è rappresentato con una transizione alto-basso al semiperiodo, il bit 0, viceversa, è rappresentato con una transizione basso-alto al semiperiodo.

> Con questa codifica si elimina il problema delle lunghe sequenze di bit con uguale valore: infatti il sincronismo tra trasmettitore e ricevitore è mantenuto grazie alle continue transizioni.

Per contro, la sua efficienza è molto inferiore rispetto alle precedenti in quanto per ogni bit da trasmettere vengono trasferiti due valori e quindi il consumo di banda è doppio. Un importante vantaggio della codifica di Manchester, tuttavia, è che permette di essere violata: il trasmettitore può emettere una sequenza di bit 1 o 0 senza effettuare la transizione, così da fornire un'informazione che il ricevitore può facilmente codificare come fine del messaggio. Un tipico esempio di **codice di violazione** è la sequenza **1100**, trasmessa senza i cambi di fronte alto-basso o basso-alto.

#prendinota
Violare il codice di Manchester vuol dire non fare la transizione a metà periodo.

#prendinota
Il codice di violazione usato dalle codifiche di Manchester (e non dalle altre codifiche) indica la fine della trasmissione.

L'ambiguità su quale rappresentazione utilizzare è superata dalla codifica detta **Differential Manchester Coding**. In questa codifica la transizione usata per codificare il dato è all'inizio del periodo invece che nel semiperiodo. Quindi una transizione all'inizio di un bit rappresenta uno 0, mentre l'assenza della transizione all'inizio rappresenta un 1. Rimane comunque, come nella normale codifica di Manchester, la transizione nel semiperiodo. Occorrerà però un segnale di riferimento iniziale per codificare il primo bit in trasmissione, come nelle modulazioni DPSK e QAM.

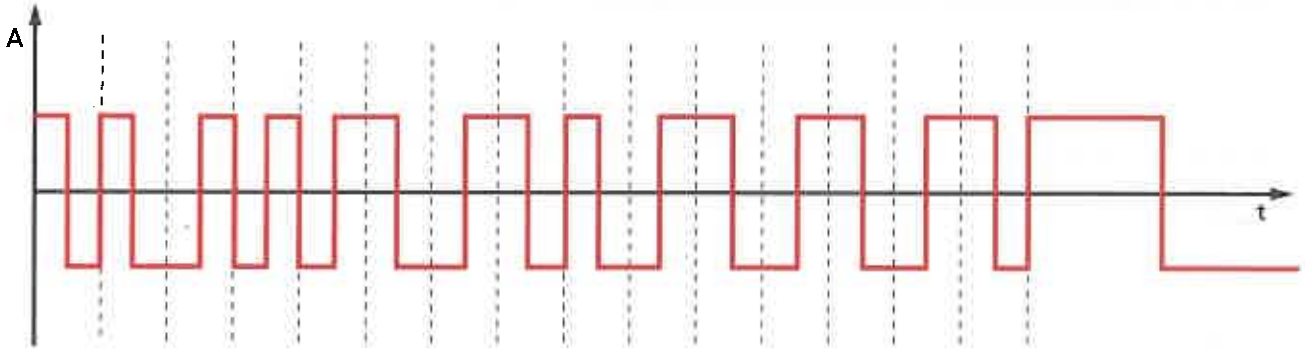
esercizio

→ PROBLEMA

Codificare secondo le regole della codifica di Manchester (nella versione di G.E. Thomas) la seguente sequenza di bit:

1100010110101011100

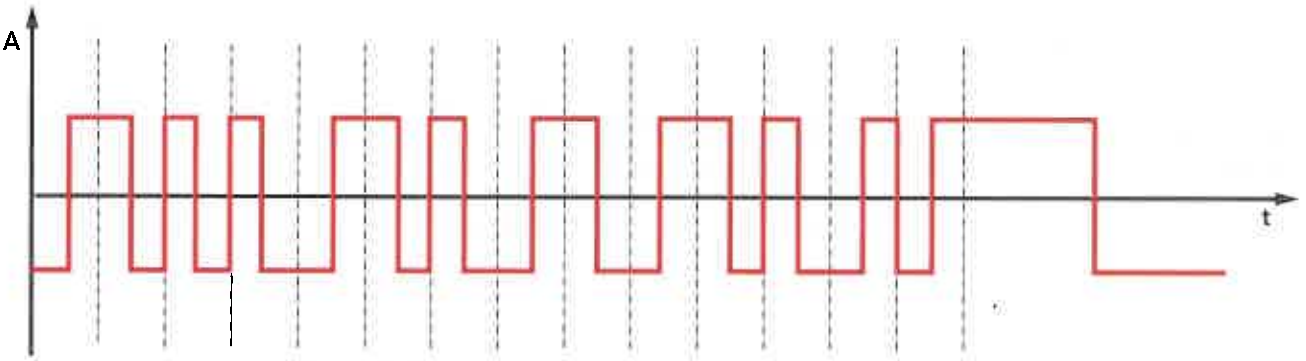
⇒ SVOLGIMENTO



esercizio

→ PROBLEMA

Data la seguente successione di bit codificati secondo la codifica di Manchester (nella versione di G.E. Thomas), determinare il flusso di bit ricevuto.



⇒ SVOLGIMENTO

La sequenza di bit ricevuti è:

0111011010110011100

3.3 Caratteristiche di una trasmissione dati

Come si è visto, una trasmissione dati consiste nell'invio di segnali da un nodo trasmettitore a un nodo ricevitore.
Gli scenari che vengono a crearsi sono molteplici, in quanto ogni trasmissione, e quindi ogni rete, si caratterizza in base a vari parametri e modalità di trasmissione.
Nel seguito sono presentati i principali.

SIMPLEX E DUPLEX

La trasmissione dati, sia essa analogica o digitale, è caratterizzata dalla direzione in cui viaggiano i segnali sui mezzi trasmissivi:

- **trasmissione simplex**: i segnali possono viaggiare in una sola direzione. Un esempio è l'uso del megafono per parlare a molte persone: la voce viaggia in una sola direzione;
- **trasmissione half-duplex**: i segnali possono viaggiare in entrambe le direzioni in un mezzo trasmissivo, ma in una sola direzione alla volta. Un esempio è il walkie-talkie;
- **trasmissione full-duplex**: i segnali possono viaggiare in entrambe le direzioni contemporaneamente; spesso queste trasmissioni sono dette bidirezionali o, semplicemente, duplex. Un esempio è la trasmissione telefonica: chiamato e chiamante possono ascoltare e parlare in simultanea usando la stessa linea telefonica.

La trasmissione full-duplex è tipica delle reti dati e spesso si associa all'utilizzo di più canali sullo stesso mezzo fisico.
Per esempio, se si utilizzano due distinti fili, uno per trasmettere e uno per ricevere, ciascun filo consente una trasmissione half-duplex, e mettendo insieme questi due fili in un cavo, si ottiene un mezzo fisico che consente una trasmissione full-duplex.

> L'impiego di mezzi trasmissivi full-duplex aumenta la velocità con cui i dati sono trasportati nella rete ed è una necessità, per esempio, per fornire un servizio di telefonia su Internet. Molti apparati di rete come i modem e le schede di rete consentono di specificare se si usa una connessione half-duplex o full-duplex.

POINT-TO-POINT E POINT-TO-MULTIPOINT

Un'altra importante caratteristica delle comunicazioni è il numero di nodi trasmettitori e ricevitori coinvolti in una stessa trasmissione. Si distinguono due casi:

- **point-to-point**: è un tipo di trasmissione che coinvolge solo due nodi, uno che trasmette e uno che riceve; questo scenario viene anche indicato come trasmissione di tipo **unicast**;
- **point-to-multipoint**: questa trasmissione coinvolge un trasmettitore e molti ricevitori e a sua volta si distingue in:
  - **broadcast**: la trasmissione avviene tra un trasmettitore e molti ricevitori sconosciuti senza preoccuparsi se il segnale trasmesso potrà essere usato dalla stazione ricevente (un esempio tipico è una stazione televisiva che trasmette un programma a migliaia di antenne riceventi; il trasmettitore non potrà sapere esattamente con chi ha comunicato, ossia chi ha ricevuto il segnale e visto il programma televisivo); questo tipo di trasmissione può essere usato sia nelle reti via cavo (wired) che senza fili (wireless) in quanto è molto semplice da realizzare e veloce;
  - **multicast**: in questo tipo di trasmissione un nodo invia i segnali a un insieme ben definito di ricevitori (per esempio un amministratore di rete decide quali workstation della rete locale possono ricevere un certo video).

IN ENGLISH PLEASE

Originally, all broadcasting was composed of analog signals using analog transmission techniques and more recently broadcasters have switched to digital signals using digital transmission

THROUGHPUT E BANDWIDTH

Un elemento molto importante nella trasmissione dati è la velocità di trasmissione, misurata in **bit per secondo (bps)**, che indica il numero di bit trasmessi in un secondo (TABELLA 1).

TABELLA 1 Unità di misura della velocità di trasmissione di una linea

Unità di misura della velocità di trasmissione	Simbolo	Equivalenza
bit per secondo	bps	Unità di misura
Kilobit per secondo	Kbps	1 Kbps = 10^3 bps = 1.000 bps
Megabit per secondo	Mbps	1 Mbps = 10^6 bps = 1.000.000 bps
Gigabit per secondo	Gbps	1 Gbps = 10^9 bps = 1.000.000.000 bps
Terabit per secondo	Tbps	1 Tbps = 10^{12} bps = 1.000.000.000 000 bps

#prendinota

Bandwidth è la velocità teorica raggiungibile mentre **throughput** è la velocità effettivamente misurata che risulta sempre essere minore della bandwidth per vari motivi: tipo di dati che devono essere trasmessi, numero di utenti della rete, topologia della rete, dispositivi di rete, disturbi presenti nel mezzo trasmissivo (il cosiddetto rumore).

La velocità di trasmissione viene espressa attraverso due concetti:

- **throughput**: è la quantità di dati (reale) che sono transitati in un canale in un certo periodo di tempo. Viene espresso come numero di bit trasmessi in un secondo. Il throughput è un'informazione utile per capire se c'è traffico in rete;
- **bandwidth** (larghezza di banda): è la quantità di dati massima (teorica) che può fluire in un canale in un dato periodo di tempo. Viene espresso come la quantità di bit trasmissibili in un secondo. La bandwidth è come il numero di corsie in autostrada: più ce ne sono e più auto possono viaggiare.

Le velocità di bandwidth e throughput variano a seconda del mezzo fisico utilizzato per la trasmissione, dato che il segnale viaggia all'interno del mezzo trasmissivo con una certa frequenza e la velocità di trasmissione è strettamente legata alla frequenza del segnale. Anche la distanza che il segnale riesce a coprire dipende dal mezzo fisico utilizzato per la trasmissione. Nell'Unità 6 affronteremo i diversi mezzi fisici di trasmissione e le loro performance in termini di velocità trasmissiva (bandwidth e throughput) e distanze coperte.

FISSA LE CONOSCENZE

- Che cosa si intende per multiplexing del canale?
- Come funziona la tecnica FDM (*Frequency Division Multiplexing*)?
- Che differenza c'è tra le codifiche Manchester e Differential Manchester?
- Spiega la caratteristica duplex della trasmissione dati (massimo 5 righe).
- Spiega la caratteristica point-to-point e point-to-multipoint della trasmissione dati (massimo 5 righe).
- Che differenza c'è tra bandwidth e throughput?

4 IL CONTROLLO DEGLI ERRORI IN TRASMISSIONE

4.1 Codici rilevatori e correttori

Il segnale inviato attraverso un canale può essere soggetto a rumore elettrico, interferenze e distorsioni che alterano il messaggio e lo rendono incomprensibile al ricevente o, peggio ancora, con un contenuto informativo differente da quello inviato dal mittente. Per fare in modo che il destinatario del messaggio sia in grado di riconoscere quando i dati ricevuti non corrispondono a quelli originali, li possa scartare richiedendone la ritrasmissione oppure possa correggerli, è necessario che il mittente aggiunga appositi codici ai bit da trasmettere, che verranno interpretati dal destinatario.

Il controllo dell'errore si basa su **codici di ridondanza** che aggiungono dei bit all'informazione da trasmettere. L'aggiunta di questi bit ridondanti consente al ricevente di verificare la correttezza dell'intera trasmissione.

Tali codici si suddividono in:

- **codici rilevatori** (error detection): in grado solo di rilevare la presenza o meno di errori nella sequenza di bit ricevuti dal destinatario, ma non la loro posizione; in questo caso il ricevente può chiedere la ritrasmissione del messaggio o segnalare l'errore all'applicazione;
- **codici correttori** (error correction): in grado di rilevare una o più posizioni errate e quindi di correggerle per semplice inversione del bit (un 1 diventa 0 e viceversa) senza che l'applicazione se ne accorga.

Quanti devono essere questi bit ridondanti (aggiunti per il controllo dell'errore) affinché il destinatario si accorga dell'eventuale errore?
Dati m bit di dati e r bit ridondanti, si ottiene un blocco complessivo di n bit ($n = m + r$), detto **codeword**, che corrisponde alla sequenza di bit trasmessa sul canale.
I codici di rilevamento/correzione degli errori si basano sul seguente principio:

i bit di ridondanza si calcolano in modo tale che non vengano utilizzate tutte le possibili codeword.

Quindi delle 2^n possibili codeword, 2^m saranno valide (cioè codeword che si possono trasmettere) e le altre indicheranno la presenza di errori. Meno sono le codeword valide, rispetto all'insieme di tutte le possibili codeword, più è possibile riconoscere e correggere gli errori.

esempio

Prendiamo una sequenza di $n = 2$ bit: questa può assumere $2^n = 2^2 = 4$ differenti configurazioni: 00, 01, 10, 11. Se tutte e 4 queste configurazioni sono usate come dati validi, un errore trasformerà una parola valida in un'altra altrettanto valida, rendendo così impossibile rilevare l'errore.

esempio

Aggiungendo alla sequenza originale un bit di ridondanza si ottengono $2^3 = 8$ configurazioni (codeword) di cui solo 4 valide. Nella TABELLA 2 sono elencate, per ogni codeword valida, le codeword errate risultanti da un errore singolo.

TABELLA 2 Codeword valide ed errate con due bit di dati e un bit di ridondanza

Stati validi	001	010	100	111
stati di errore	000	000	000	110
stati di errore	011	011	110	011
stati di errore	101	110	101	101

Con l'aggiunta di un bit e la scelta delle codeword ammesse gli stati di errore non possono essere interpretati come stati validi; risulta così semplice la rilevazione dell'errore (singolo). Inoltre ogni codeword errata differisce dalla corrispondente valida per un solo bit, mentre due configurazioni valide differiscono tra loro per 2 bit (per esempio 010 e 111 differiscono nel primo e nell'ultimo bit).

DISTANZA DI HAMMING
Misura il numero di sostituzioni necessarie per convertire una codeword valida in un'altra codeword valida

Date due codeword valide si definisce **distanza di Hamming** tra esse il numero di bit di cui differiscono a parità di posizione. Dato un codice a n bit, si definisce **distanza di Hamming del codice** la distanza di Hamming **minima** tra tutte le codeword del codice stesso.

Ciò significa che, se due codeword hanno distanza di Hamming pari a **d**, saranno necessari **d** errori di singoli bit per trasformare una codeword valida in un'altra valida, ingannando il ricevente.
La proprietà di un codice di rilevare/correggere gli errori dipende dalla sua distanza di Hamming. Infatti, vale quanto segue:

- **per rilevare k errori** è necessario un codice la cui distanza sia **d = k + 1** perché è impossibile che k errori di singoli bit trasformino una codeword valida in un'altra codeword valida: ne servono k +1;
- **per correggere k errori** è necessario un codice con una distanza **d = 2k + 1**, perché in esso le codeword valide sono così distanti che, anche se si verificassero k alterazioni di bit, la codeword originale risulterebbe più vicina a quella alterata che a qualunque altra, per cui sarebbe univocamente determinabile.

Nel primo esempio la distanza di Hamming è 1:

d = k + 1; con **d = 1** si ha **k = 0**

non è possibile rilevare alcun errore.
Nel secondo esempio la distanza di Hamming è 2:

d = k + 1; con **d = 2** si ha **k = 1**

è possibile rilevare un errore singolo, ma non correggerlo, infatti:

d = 2k + 1; con **d = 2** si ha **k = 0**

Se vogliamo che un codice sia in grado di rilevare errori (singoli e doppi) e correggere automaticamente errori singoli, occorre che la distanza di Hamming minima sia 3:

d = k + 1; con **d = 3** si ha **k = 2** → **rileva fino a errori doppi**
d = 2k+1; con **d = 3** si ha **k = 1** → **corregge errori singoli**

Poter correggere un errore automaticamente significa evitare la ritrasmissione della stringa: il ricevente individua l'errore e lo corregge da sé.
Naturalmente questo ha un costo in termini di un maggior numero di bit da trasmettere e, quindi, una maggiore occupazione del canale trasmissivo.
Possiamo sintetizzare con le seguenti implicazioni:

distanza di Hamming alta → tanti bit ridondanti → codeword più lunghe → correzione automatica errori

#prendinota

Per poter correggere automaticamente un errore singolo occorre sapere la posizione del bit sbagliato all'interno dalla stringa. I codici correttori riescono a fare questo.

Vediamo le tre principali tecniche di correzione dell'errore in trasmissione mediante codici rilevatori/correttori:

- codici di parità (solo rilevatore errori singoli);
- codici di ridondanza ciclica (solo rilevatore);
- codici di Hamming (rilevatore e correttore).

4.2 Codici di parità

I codici di parità sono quelli in cui la distanza di Hamming è 2 e sono quindi in grado di rilevare la presenza di un **errore singolo** (senza poterlo correggere perché non ne determinano la posizione) o, più in generale, rilevano l'occorrenza di un numero **dispari** di errori. Alla sequenza di bit da trasmettere si aggiunge un bit di controllo in modo che il numero totale (bit di dati più il bit di controllo) di "1" sia pari (*parità pari*) oppure dispari (*parità dispari*) a seconda del protocollo scelto.

esempio

Sia data la sequenza di bit: 01100010101111

Il numero di bit "1" è 8, quindi pari, allora il bit di parità sarà 0 per parità pari e 1 per parità dispari:

parità pari: 011000101011110

parità dispari: 011000101011111

Il ricevitore provvederà a ricalcolare il bit di parità sulla sequenza di bit ricevuta, escluso il bit di parità aggiunto, e confronterà il bit di parità ottenuto con quello ricevuto: se sono diversi la trasmissione non è avvenuta correttamente, se invece sono uguali è probabile che la sequenza ricevuta sia quella originale; tuttavia, poiché questa tecnica non rileva gli errori doppi, non se ne ha la garanzia assoluta.

Un altro problema che i codici di parità presentano è che non sono in grado di riconoscere quando l'errore è sul bit di parità: questo comporta il rilevamento di un errore quando, invece, la sequenza dei bit dati è stata ricevuta correttamente.

Per questi problemi, l'utilizzo dei codici di parità è limitato ai trasferimenti all'interno del computer (tramite la motherboard). Per esempio i bus SCSI e PCI usano la parità per trovare errori di trasmissione, inoltre molte memorie cache includono tale sistema di controllo e correzione. Dato che nelle cache i dati sono solo una copia di quelli nella RAM, se vi si trova un errore, può essere ricaricata.

5 IL CONTROLLO DI FLUSSO

5.1 La tecnica Stop and Wait

Nella trasmissione dati tra un mittente e un destinatario è necessario regolare il flusso dei dati per evitare che il mittente invii i dati a una velocità superiore alla capacità di ricezione del destinatario (dimensione del buffer di ricezione, velocità di elaborazione, ecc.), con il rischio di perdita di informazioni.
In generale, i protocolli per il **#controllo di flusso** prevedono che il destinatario trasmetta un messaggio di riscontro della corretta ricezione del messaggio, detto **acknowledge (ACK)**.

#techwords

Il **controllo di flusso** (*flow control*) è l'insieme dei meccanismi che consentono di regolare la velocità di trasmissione dei dati in modo che il destinatario riesca a elaborare quanto riceve.

Questo meccanismo prende il nome di **Stop and Wait** e prevede che il mittente fermi la trasmissione dopo ogni invio di dati e attenda l'ACK di riscontro della corretta ricezione di quanto inviato prima di riprendere a trasmettere.

Il messaggio di riscontro (ACK) viene inviato dal ricevitore solo se il messaggio è stato ricevuto senza errori, in caso contrario quest'ultimo verrà scartato e l'ACK non verrà inviato. Il trasmettitore imposta un **timeout** all'invio di ogni messaggio: se al suo scadere non avrà ricevuto l'ACK, ritrasmetterà il messaggio.

La tecnica dello Stop and Wait viene anche rafforzata dalla presenza di un **numero di sequenza** che assume alternativamente il valore 0 e 1. Esso serve a individuare i messaggi duplicati: infatti se l'ACK venisse perso a causa del rumore di linea, il mittente invierebbe di nuovo l'ultimo messaggio. Il numero di sequenza permette al destinatario di scartare il messaggio se ha lo stesso numero di sequenza di quello precedentemente ricevuto (il ricevitore non può ricevere due 0 o due 1 di fila).

Si presentano quindi tre possibili scenari (FIGURA 21):

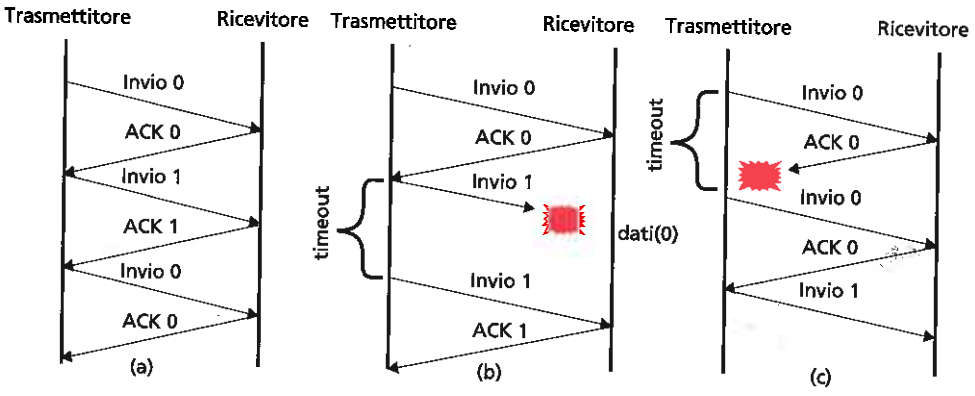


FIGURA 21 La tecnica Stop and Wait

1. **trasferimento corretto**: a ogni trasmissione di una sequenza di bit c'è un riscontro che conferma l'avvenuta ricezione;
2. **dati non arrivati**: la sequenza Invio 1 trasmessa non è arrivata al destinatario (o è arrivata errata e quindi è stata scartata): allo scadere del timeout il mittente ritrasmette la stessa sequenza di dati;

3. **riscontro non arrivato**: nel caso in cui la sequenza inviata arrivi al destinatario ma il suo messaggio di ACK non arrivi al mittente (o arrivi errato), allo scadere del timeout il mittente provvede a ritrasmettere la sequenza; il destinatario, controllando il numero di sequenza, riconosce che si tratta di un messaggio duplicato e lo scarta, inviando, però, nuovamente il messaggio di conferma ricezione (ACK).

In questo meccanismo un elemento critico è il timeout del mittente: infatti se troppo corto potrebbero venire ritrasmessi messaggi che invece erano stati inviati e ricevuti correttamente (duplicazione), mentre se troppo lungo risultano aumentati i tempi di trasmissione con conseguente basso utilizzo della banda disponibile.

5.2 La tecnica a finestra

Questa tecnica prevede di non inviare il riscontro a ogni messaggio ricevuto, ma consentire che venga trasmesso un certo numero di messaggi (**burst**, raffica) fino a un massimo prefissato **w** che rappresenta la *dimensione della finestra* (**window size**) prima di inviare l'ACK a riscontro dell'intera finestra di invii e non più del singolo invio. Raramente è possibile avere il canale a disposizione per trasmettere in sequenza un'intera finestra di dimensione w. La dimensione w rappresenta di solito un valore massimo mentre il numero di invii in burst senza l'attesa dell'ACK è tipicamente minore di w. La FIGURA 22 mostra un esempio di tecnica a finestra con dimensione della finestra w = 7 (dall'invio 0 all'invio 6) in cui il mittente riesce a inviare fino a 3 messaggi per volta. Il trasmettitore effettua 3 invii senza attendere la conferma della loro ricezione (finestra aperta). Il ricevitore invia ACK 3 a indicare che gli invii 0, 1 e 2 sono stati ricevuti correttamente e attende di ricevere il numero 3 (finestra aperta). Il trasmettitore effettua altri 3 invii e il ricevitore invia ACK 6 a indicare che gli invii 3, 4 e 5 sono stati ricevuti correttamente e attende di ricevere il numero 6 (finestra aperta). Infine il trasmettitore effettua un altro invio, dopodiché deve interrompere la trasmissione e aspettare di ricevere un riscontro (finestra chiusa) avendo fatto tutti e sette gli invii previsti dalla finestra.

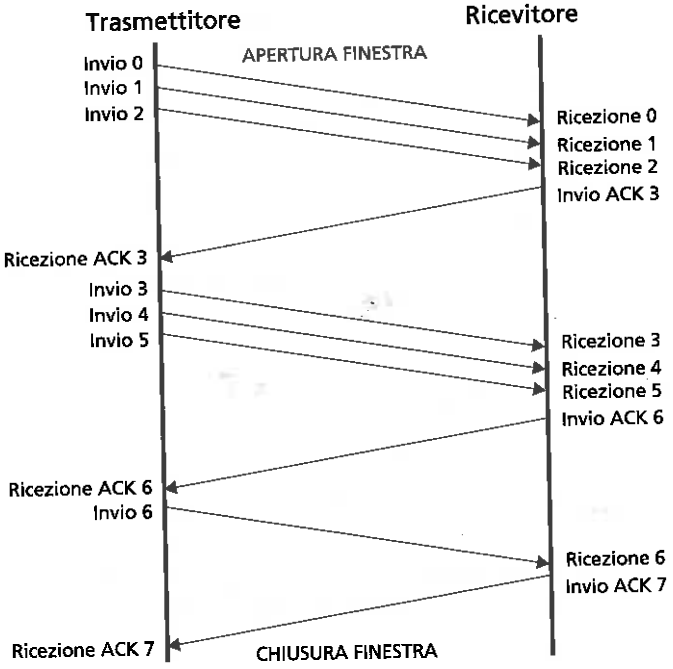


FIGURA 22 La tecnica a finestra

#prendinota

Nella tecnica a finestra i riscontri, **ACK**, contengono il numero di sequenza del **successivo** messaggio atteso.

6 I SISTEMI APERTI: PROTOCOLLI E STANDARD

6.1 I sistemi aperti

La #comunicazione e le #reti sono ormai entrate a far parte del nostro quotidiano. La rete telefonica, i sistemi di posta elettronica, Internet, i social network, le app per i servizi online sono utilizzate da molte persone nel mondo tutti i giorni.
Vari tipi di sistemi permettono di connettersi a una rete: dai computer ai telefoni, dalla televisione alle console dei giochi. Lo scopo principale è di condividere dati, risorse e, più in generale, di **comunicare**.
Raramente due computer che devono comunicare sono connessi direttamente: spesso si trovano a una distanza tale da rendere impossibile collegarli tramite un cavo o anche in modalità wireless, cioè "senza fili".
Altra situazione è quella in cui ci sono più computer, ognuno dei quali si deve connettere agli altri: anche in questo caso non è praticabile avere un collegamento diretto tra tutti i computer che devono comunicare.
La soluzione è di connettere ogni computer a una **rete di comunicazione**.

#techwords

La comunicazione tra due sistemi è il processo che consente a essi di scambiarsi delle informazioni.

#techwords

Una rete è un insieme di nodi (elaboratori o apparati) connessi tra loro da canali (linee di comunicazione).

Le reti di calcolatori di prima generazione nacquero e si svilupparono come **sistemi chiusi**, il che significava che *per le telecomunicazioni* ci doveva essere una rete specializzata per ogni tipologia di servizi (per esempio la rete per la telefonia non trasportava dati, e viceversa) e *per l'informatica* significava che tutte le macchine della rete dovevano appartenere allo stesso costruttore (*vendor*).

Questa situazione, se garantiva un guadagno ai costruttori, non era un vantaggio per gli operatori di rete e per gli utenti finali, che erano costretti a realizzare e utilizzare reti *mono-vendor*. Infatti:

- gli apparati di un costruttore non riuscivano a interpretare i segnali dei dispositivi di altri costruttori;
- nel momento in cui i computer riuscivano a connettersi fisicamente, non potevano colloquiare tra loro perché parlavano linguaggi diversi (per esempio usavano differenti Sistemi Operativi);
- i programmi applicativi non riuscivano a operare in un ambiente distribuito (cioè suddiviso su più macchine in luoghi remoti).

Con l'evolvere della tecnologia e con le pressioni degli utenti finali, nacquero degli enti di standardizzazione internazionale che produssero una serie di norme tecniche per le telecomunicazioni in rete, avviando il progetto per la definizione di un modello di **sistema aperto** per l'interconnessione di reti di computer che prescindesse dal microprocessore scelto, dal Sistema Operativo installato o in generale dall'hardware e dal software utilizzati.

L'obiettivo era di avere reti *multi-vendor* in cui qualunque computer o apparato di rete fosse in grado di comunicare con qualunque altro computer o apparato, mediante una qualunque rete.
Per realizzare sistemi aperti è necessario stabilire delle regole comuni per lo scambio delle informazioni, quindi si devono definire dei **protocolli** e degli **standard**.

6.2 I protocolli

Nelle reti la comunicazione avviene tramite dispositivi in grado di trasmettere e ricevere informazioni. Chiaramente affinché il messaggio inviato sia *compreso* da chi lo riceve, è necessario che mittente e destinatario si accordino sulle modalità di trasferimento dei dati («Sei pronto a ricevere il mio messaggio?», «Sto per inviarti il messaggio», «Hai ricevuto il messaggio che ti ho inviato?», «Hai rilevato problemi in rete dalla tua parte?», ecc.) e su come deve essere costruito il messaggio. Infatti il destinatario deve saper interpretare secondo certe regole stabilite i dati ricevuti, altrimenti questi non sarebbero altro che sequenze di bit senza significato («L'indirizzo del mittente è scritto dal 21° bit al 52° bit, quello del destinatario dal 53° bit all'84° bit»).
Da qui nasce l'esigenza di definire dei protocolli di comunicazione che governino il trasferimento dei dati, stabilendo come e quando le informazioni devono essere comunicate.

Un protocollo è un insieme di regole descritte in modo formale che vengono stabilite al fine di realizzare la comunicazione tra due o più entità. Un protocollo definisce come è codificata l'informazione (formato del messaggio) e le azioni da intraprendere in seguito alla trasmissione/ricezione di un messaggio o di altri eventi (per esempio in caso di errore si deve prevedere un'adeguata reazione).

La definizione di un protocollo si compone di tre parti:

- la **sintassi**: descrive come sono strutturati i dati (ossia l'ordine con cui si presentano);
- la **semantica**: descrive il significato delle varie sequenze di bit, consentendo al destinatario di interpretare correttamente ciò che ha ricevuto e di comportarsi di conseguenza;
- la **sincronizzazione**: descrive quando i dati vanno inviati, definendo sequenze temporali di emissione dei comandi e delle risposte.

I protocolli sono gestiti dal software di rete del dispositivo, gli utenti finali devono solo preoccuparsi dei dati che vogliono trasmettere e non dei protocolli utilizzati.

esempio

Per comprendere meglio il significato di protocollo, si può prendere come esempio il protocollo "umano" usato per inviare una lettera tramite il servizio postale e confrontarlo con quello usato dall'applicazione di posta elettronica del computer per inviare un messaggio (FIGURA 24):

- l'indirizzo scritto sulla busta corrisponde al campo destinatario del messaggio (nell'esempio mario.rossi@gmail.com);

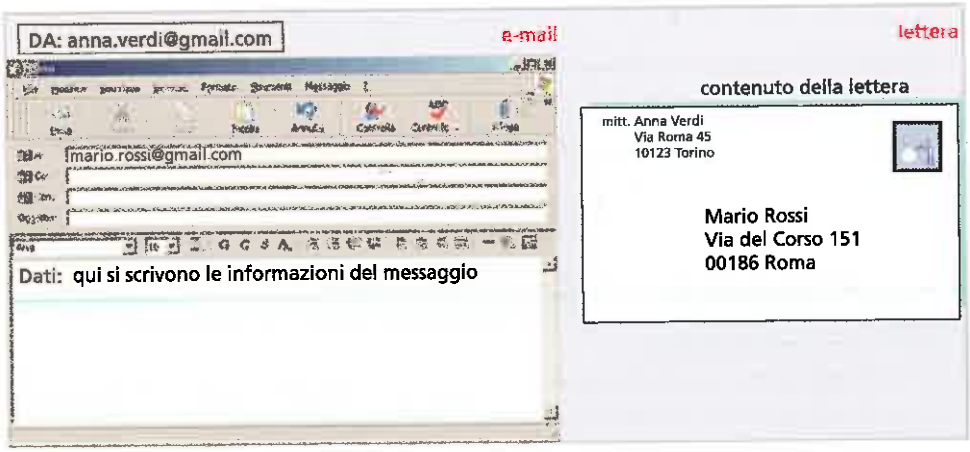


FIGURA 24 Confronto fra posta elettronica e posta tradizionale

- il mittente scritto sulla busta corrisponde al campo sorgente del messaggio (nell'esempio anna.verdi@gmail.com);
- il contenuto della busta corrisponde alle informazioni del messaggio.

Se non si seguono le regole stabilite dal protocollo per l'invio di una lettera tramite il servizio postale, per esempio non si scrive correttamente il campo indirizzo, essa non sarà ricevuta. Analogamente se l'indirizzo del computer destinatario del messaggio non è corretto, le informazioni inviate non arriveranno a destinazione.

Tra i principali protocolli per le reti che studieremo ci sono:

- TCP (Transmission Control Protocol) e IP (Internet Protocol) che danno il nome al modello **TCP/IP**, su cui si basa il funzionamento di **Internet**, la rete delle reti;
- il protocollo per il web HTTP (HyperText Transfer Protocol;
- il protocollo per la posta elettronica SMTP (Simple Mail Transfer Protocol);
- il protocollo per il trasferimento file FTP (File Transfer Protocol).

6.3 Gli standard

La comunicazione tra due entità avviene quindi secondo un protocollo stabilito e noto a entrambe. Cosa succede se, per esempio, i due dispositivi sono di produttori diversi che utilizzano nel loro software di rete protocolli diversi? La comunicazione diventa impossibile.

È quindi necessario far riferimento agli **standard internazionali** che consentono di avere un mercato aperto e competitivo (posso inserire nel mio computer la scheda di rete che preferisco, l'aderenza allo standard mi garantisce la comunicazione con qualunque altro computer in rete).

Gli standard sono importanti in qualunque campo, ma diventano indispensabili nei sistemi di comunicazione che consistono di molti componenti diversi che devono lavorare insieme. Questo obiettivo, di per sé ovvio, in realtà è spesso arduo da raggiungere, in quanto è difficile mettere d'accordo produttori di apparati, gestori di rete, fornitori di servizi, ognuno con i propri interessi economici, nel definire standard a cui tutti si dovranno adeguare per garantire l'interoperabilità tra dispositivi e reti eterogenee.

> Uno **standard** fornisce le linee guida a cui ci si deve adeguare per assicurare il livello di interconnessione necessario per realizzare comunicazioni in ambito locale e internazionale.

Nell'ambito delle telecomunicazioni esistono molti standard che vengono costantemente aggiornati o se ne creano di nuovi per riflettere le nuove forme di comunicazione, basti pensare all'evoluzione che nei primi anni Duemila ha portato al diffondersi delle reti senza fili o all'utilizzo delle porte USB.

FIGURA 25 Logo dello standard Wi-Fi e dello standard USB

FISSA LE CONOSCENZE

- Che cos'è un sistema aperto?
- Spiega il significato di protocollo.
- Spiega il significato di standard.

7 LA CONDIVISIONE IN RETE

L'esigenza di disporre di una rete (FIGURA 26) nasce dalla necessità di comunicare e scambiare informazioni velocemente. Ma uno dei maggiori vantaggi della rete è quello della **condivisione**.

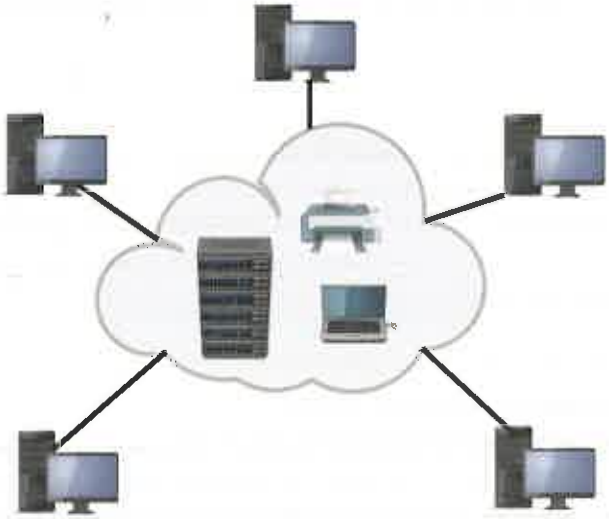


FIGURA 26 Esempio di rete che condivide risorse

In tutte le aziende si lavora in team su progetti di cui ognuno svolge una parte. Questo implica che gli aggiornamenti delle informazioni devono essere tali da essere immediatamente fruibili da tutti. Il rischio di avere copie diverse degli stessi dati risulta molto ridotto se i dati si condividono in tempo reale.

> Per condividere serve una rete affinché tutti vi accedano per caricare o scaricare le informazioni che di volta in volta necessitano.

La disponibilità di una rete ha quindi numerosi vantaggi riguardo alla condivisione:

- si possono usare stampanti collegate ad altri computer;
- si possono condividere i file del proprio computer con altri computer e viceversa;
- si può accedere a drive online condivisi;
- si possono usare programmi e software esistenti in remoto;
- si possono usare dati (DATABASE) esistenti su altri computer;
- si può condividere l'accesso a Internet tramite un unico dispositivo.

Questi vantaggi a loro volta comportano il miglioramento di alcune caratteristiche dell'azienda:

- minori costi dovuti alla condivisione delle risorse di rete (dati, stampanti, periferiche varie, software);
- accesso alle informazioni e ai dati in tempo utile;
- comunicazione e organizzazione più efficaci.

La sempre maggior diffusione dello **smart working** (il lavoro da remoto, a casa anziché in ufficio) ha reso indispensabile la condivisione delle risorse aziendali e la necessità di accedervi da ovunque.

Vediamo i principali esempi di condivisione di risorse in una rete.

CONDIVISIONE DELLE STAMPANTI

Se non abbiamo a disposizione una rete, è necessario avere una stampante per ogni computer oppure spostare la stampante da un computer all'altro.
Se abbiamo a disposizione una rete informatica invece serve una sola stampante, collegata alla rete e condivisa con tutti gli utenti della rete. Ogni utente della rete può stampare liberamente: la stampante metterà in coda le stampe man mano che arrivano dagli utenti. Questo permette di ottimizzare risorse e costi.

CONDIVISIONE DI FILE E DATI

Ogni computer può mettere a disposizione i suoi file e condividerli con altri utenti della rete. Potenzialmente tutti i computer possono mettere a disposizione i propri dati, creando così una rete di dati distribuita. Diversamente, si possono mettere a disposizione i dati su un unico computer, cui possono accedere tutti gli utenti: in questo caso si parla di rete di dati centralizzata.

CONDIVISIONE DELL'HARDWARE E DEI SERVIZI

La nascita di elaboratori sempre più performanti ha reso possibile condividere, su un unico computer, servizi per le aziende. Questo è all'origine del **cloud computing**, con il quale è possibile fornire l'hardware, il software o entrambi in remoto a quelle aziende (startup, Pubblica Amministrazione) che difficilmente potrebbero investire in elaboratori potenti e costosi e garantirne l'aggiornamento e la manutenzione.

CONDIVISIONE DEL SOFTWARE

I software possono essere progettati per funzionare in rete. È possibile quindi installare il software su un solo computer, così che tutti gli utenti della rete possano usare il programma senza doverlo installare sul proprio computer. Questo vale sia per chi lavora in ufficio sia per chi lavora in smart working.

CONDIVISIONE DI INTERNET

Per una rete privata aziendale (ma anche domestica) è molto meglio avere un solo collegamento alla rete pubblica Internet (salvo una seconda linea per eventuali guasti sul collegamento principale). La disponibilità della rete permette di impostare il collegamento Internet su un solo dispositivo e di condividerlo con tutti gli utenti (i computer) della rete pagando un solo abbonamento al gestore di servizi telefonici e Internet (**ISP**, Internet Service Provider) scelto.
La condivisione di quel collegamento con tutti i computer della rete privata comporta quindi un risparmio economico, ma anche una maggior facilità di monitoraggio e protezione dei dati e delle comunicazioni avendo un solo punto di ingresso/uscita da controllare.

FISSA LE CONOSCENZE

- Perché per condividere serve una rete?
- La condivisione in rete che miglioramenti apporta al lavoro aziendale?
- A chi conviene poter condividere hardware e servizi?
- Perché conviene condividere l'accesso a Internet in una rete privata?

8 I PARADIGMI CLIENT-SERVER E PEER-TO-PEER

8.1 Il modello Client-Server

Le reti di computer sono formate da macchine in grado di lavorare in autonomia e collegate tra loro in modo da potersi scambiare informazioni.
Le reti possono essere realizzate secondo paradigmi diversi. I più usati al giorno d'oggi sono il Client-Server e il Peer-to-Peer.
Il **Client-Server** è il paradigma più utilizzato sia nella rete Internet che nella gran parte delle reti aziendali: è quindi il più diffuso.

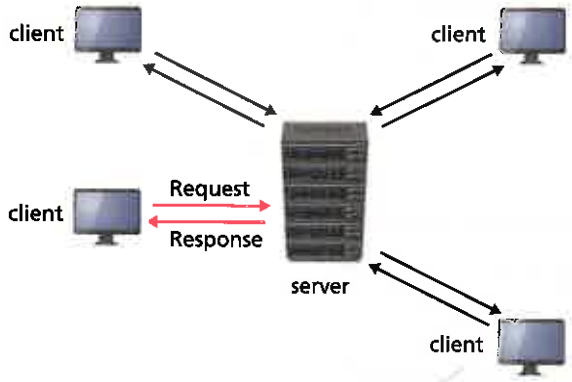


FIGURA 27 Modello Client-Server

Ogni servizio applicativo offerto ha una componente *client* e una *server* (FIGURA 27):

- **server**: è un processo che offre un servizio e può essere raggiunto attraverso la rete; è in grado di accettare le richieste che gli arrivano dai client, elaborarle, effettuare il servizio richiesto e restituire il risultato al richiedente (o un messaggio di errore se non è riuscito a soddisfare la richiesta); solitamente il processo server viene avviato all'accensione del computer e rimane sempre attivo;
- **client**: è un processo che invia una richiesta a un server e resta in attesa della risposta; tipicamente diventa attivo quando deve inviare una richiesta e, una volta ricevuta la relativa risposta, diventa inattivo.

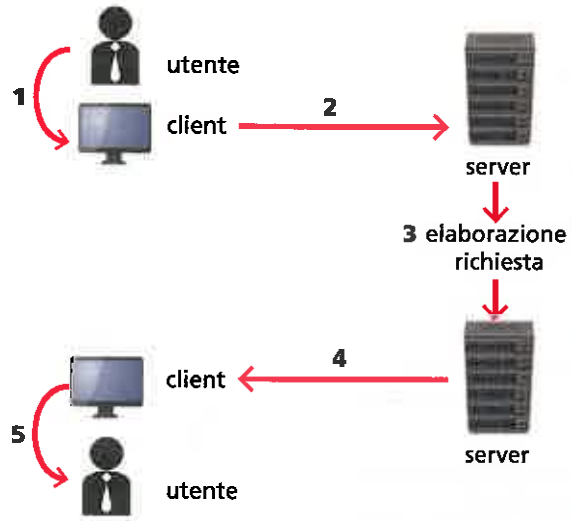


FIGURA 28 I passaggi dell'intero processo nel modello Client-Server

La FIGURA 28 riassume i passi dell'intero processo.
Esempi di applicazioni Client-Server di uso comune su Internet sono:

- la navigazione sul web in cui il client è il browser che chiede una pagina web e il web server è il server che risponde;
- la posta elettronica in cui il client è il programma di posta elettronica che effettua l'invio e il server di posta è il server che risponde;
- il download di un file da un sito in cui il client è l'applicativo che chiede il download e il file server è il server che risponde.

Anche in ambito locale, nelle reti aziendali, il paradigma Client-Server è usato ad esempio per:

- fornire un servizio di stampa condiviso tra i client della rete;
- assegnare un indirizzo ai client collegati;
- fornire il collegamento a Internet ai client che lo richiedono.

Le reti che applicano questo modello necessitano di un amministratore di rete che si occupi della gestione del server, di effettuare il backup dei dati e delle politiche di sicurezza. Le reti aziendali adottano questo modello in quanto offre ottime garanzie di sicurezza.

8.2 Il modello Peer-to-Peer (P2P)

Il **Peer-to-Peer** è un modello di applicazione distribuita in cui le applicazioni fungono sia da server che da client. Le reti che applicano tale modello non prevedono la distinzione tra computer server e computer client: ogni computer ha un ruolo paritetico rispetto agli altri. È un rapporto tra pari (**peer**) e non più tra un client che chiede e un server che risponde. I singoli utenti sono i responsabili delle risorse del proprio computer e possono decidere in autonomia quali risorse (hardware o software) condividere. Questa caratteristica comporta l'assenza di un amministratore di rete con la conseguenza che la sicurezza viene demandata al singolo utente: non esiste una politica comune, ma ogni computer della rete decide la propria politica di protezione dei dati.
Questo modello si applica a reti piuttosto piccole, con non più di 10 computer. Tutto è iniziato con Napster, un'applicazione creata nel 1999 per diffondere contenuti, soprattutto file MP3. Esempi di programmi P2P molto usati sono JDownloader, eMule e le applicazioni della rete BitTorrent.

Vale la pena evidenziare che i confini tra Client-Server e Peer-to-Peer non sono poi così netti: infatti il modello P2P può essere costruito sopra il modello Client-Server ed entrambi possono essere usati come base per applicazioni tradizionali o nuove.
In generale, però, si possono evidenziare alcuni vantaggi tipici delle reti P2P e delle reti Client-Server, come mostrato nella TABELLA 5.

Reti Peer-to-Peer		Reti Client-Server	
vantaggi	**svantaggi**	**vantaggi**	**svantaggi**
Non richiede un amministratore di rete	L'utente deve imparare alcuni compiti di gestione della rete	Amministrazione centralizzata, utile per grandi reti	Richiede la figura professionale di un amministratore di rete
Non richiede software specifico per l'amministrazione della rete	Poco sicura	Fornisce un buon grado di sicurezza	Costi più alti per un software specifico per la gestione e l'operatività della rete
Poco costosa	Le prestazioni dei computer che condividono le risorse possono diminuire significativamente	Tutti i dati sono salvati su un computer centrale	Richiede una piattaforma hardware potente (e costosa)
	Non si adatta a grandi reti (ingestibili senza un amministratore)		Il server è un "single point of failure", ossia se non funziona i dati non sono accessibili

TABELLA 5 Vantaggi e svantaggi delle reti Peer-to-Peer e Client-Server

FISSA LE CONOSCENZE

- Quali sono i compiti del server e quali quelli del client?
- Quale tra i due paradigmi studiati è il più utilizzato e perché?

9 CLASSIFICAZIONE E TOPOLOGIA DELLE RETI LAN, MAN E WAN

9.1 La classificazione delle reti in base all'estensione

Le reti evolvono continuamente sia come progettazione che come utilizzo, diventando così sempre più complesse.
Nel nostro studio delle reti scopriremo come esse si possono distinguere e classificare in vari modi a seconda della caratteristica e funzionalità di interesse: in base a come sono organizzate, che tipo di dati trasportano, quali apparati di rete sono utilizzati, con quali mezzi fisici sono realizzati i collegamenti, ecc. Uno dei modi più utilizzati per classificare le reti è basato sulla loro estensione, quindi sull'area che sono in grado di coprire.

LOCAL AREA NETWORK (LAN)

Si tratta di reti non molto grandi, la cui estensione è confinata in un edificio o in un campus, senza attraversare suolo pubblico. Col tempo questa definizione è però evoluta verso un concetto meno fisico e più amministrativo: una LAN è un insieme di reti interconnesse che risulta essere sotto il controllo di un solo gruppo amministrativo che si occupa, in modo particolare, di gestirne la sicurezza in termini di controllo dell'accesso alla rete e delle operazioni che possono essere svolte tramite essa.

METROPOLITAN AREA NETWORK (MAN)

È una rete che copre l'area di una città o di una provincia (città metropolitana) o di una piccola regione e opera a velocità che sono paragonabili con quelle delle LAN. Il suo utilizzo è molto diffuso nella Pubblica Amministrazione per la realizzazione di servizi in ambito comunale (ospedali, biblioteche, ecc.).

WIDE AREA NETWORK (WAN)

È una rete estesa geograficamente, che connette LAN e MAN sparse nel mondo. Poiché i nodi possono essere collegati anche a grandi distanze (migliaia di chilometri) vengono di norma utilizzati mezzi di comunicazione poco costosi e già ampiamente diffusi (generalmente le linee telefoniche) con la conseguenza che la trasmissione può risultare più lenta. Attualmente i gestori di servizi telefonici e telematici si stanno indirizzando verso l'impiego delle fibre ottiche in sostituzione dei cavi elettrici, così da rendere la trasmissione più veloce.

#techwords

Un **host** è un generico dispositivo della rete. Può essere di volta in volta un personal computer o un dispositivo mobile, un server o un client, una workstation o una stampante, ma anche dispositivi per la comunicazione tra reti come un router o uno switch che studieremo nella prossima unità.

9.2 La topologia delle reti

La topologia definisce la struttura di una rete, cioè di come sono posizionati i nodi (elaboratori e apparati) che chiameremo genericamente **#host** (ospite) collegati tra loro.

TOPOLOGIA A BUS

Questa topologia usa un singolo backbone (linea principale), detto **bus**, a cui si collegano tutti gli host; alle due estremità del cavo è collocata una resistenza terminale, detta terminatore (FIGURA 29).

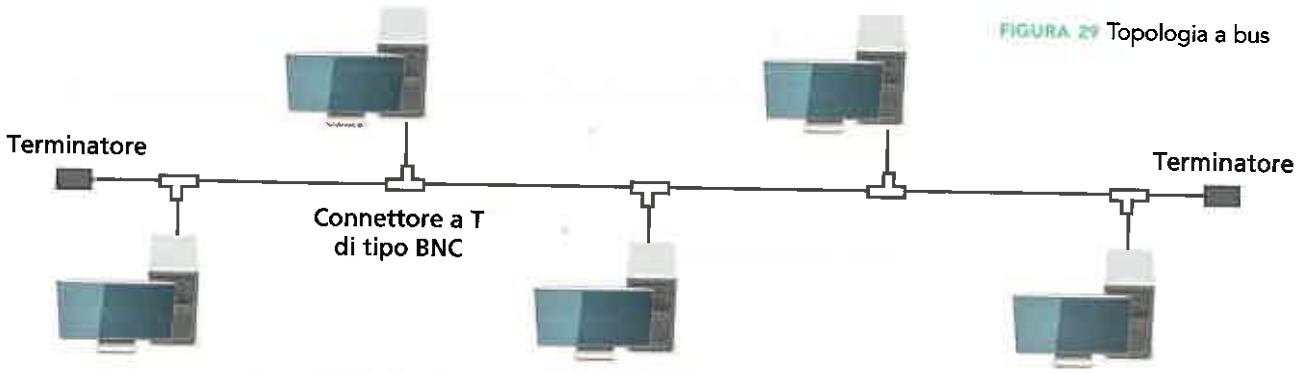


FIGURA 29 Topologia a bus

Poiché esiste un solo percorso possibile tra ogni coppia di nodi, è una topologia a basso costo.
I segnali passano lungo i cavi tra i due terminatori e vengono controllati da tutti gli host attestati sul bus: solo se l'indirizzo di destinazione del messaggio coincide con quello dell'host, il messaggio viene ricevuto ed elaborato dall'host. Si tratta quindi di una trasmissione di tipo broadcast cioè inviata a tutti.

Se un host non funziona la rete continua a funzionare, ma se si guasta il cavo verso l'host l'intero bus compreso tra i due terminatori smette di funzionare. Il cavo è quindi il punto debole di questa topologia in quanto un guasto su di esso provoca il malfunzionamento dell'intera rete.

Inoltre, il fatto di avere un unico canale condiviso implica il non poter avere due trasmissioni in contemporanea.
Questa topologia è tipica delle reti locali e metropolitane, molto usata in passato non viene più realizzata per la sua bassa tolleranza ai guasti.

TOPOLOGIA AD ANELLO

In questa topologia, detta anche **ring**, un cavo collega un host al precedente e al successivo creando un circuito di rete continuo su cui sono trasmessi i dati (FIGURA 30).

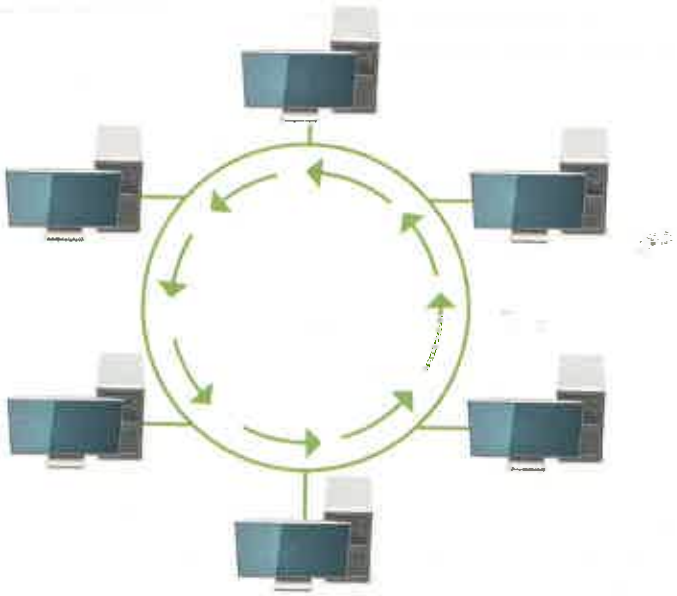


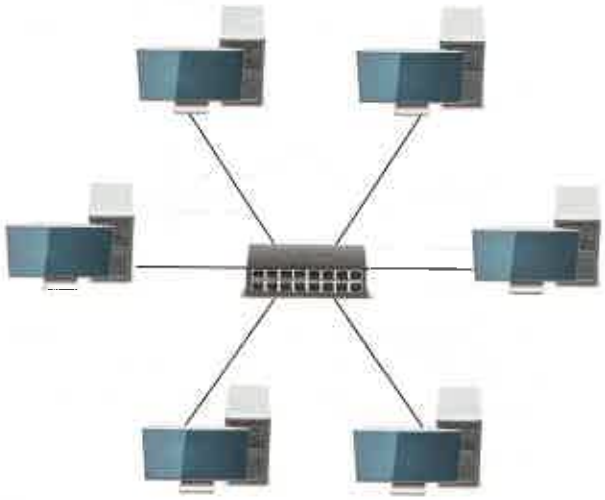
FIGURA 30 Topologia ad anello

La topologia ad anello è unidirezionale ma può avere un anello secondario in direzione inversa che entra in funzione se il primario si guasta. In caso di guasto, l'anello secondario consente alla rete di continuare a funzionare, anche se con una capacità ridotta. Questa topologia è usata nelle reti locali e, soprattutto, in quelle metropolitane.

TOPOLOGIA A STELLA

In questa topologia tutti gli host sono collegati a un punto centrale, detto **centro stella**, che di solito è un apparato di rete (hub, switch o router) e costituisce il punto di connessione comune in modo che i computer siano in comunicazione l'uno con l'altro (FIGURA 31).

FIGURA 31 Topologia a stella (LAN)



Anche se questa topologia porta a un aumento del numero dei cavi rispetto, per esempio, a quella a bus, essa offre notevoli vantaggi in termini di:

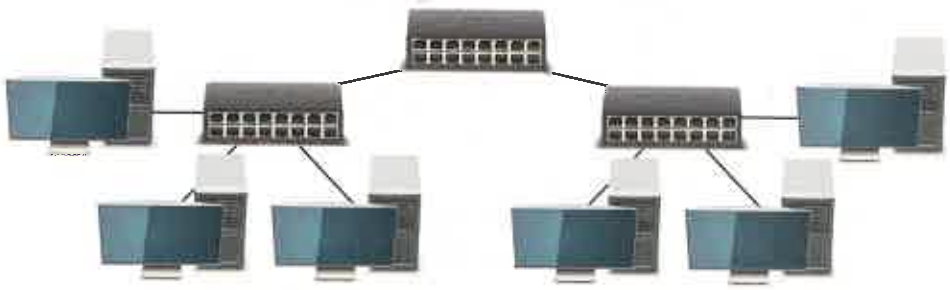
- fault-tolerance (tolleranza ai guasti): il guasto di un canale o nodo della rete non ne compromette il funzionamento generale;
- flessibilità ed espandibilità: infatti lo spostamento di un host da un punto a un altro della rete o l'inserimento di uno nuovo non richiedono il fermo della rete;
- semplicità di gestione.

Per contro è vulnerabile nel centro della stella: se l'apparato che svolge questo ruolo si guasta, la rete smette di funzionare.
Questa topologia è usata nelle reti locali, nelle reti satellitari e in quelle radio.

TOPOLOGIA A STELLA ESTESA

Questa topologia, detta anche a **gerarchica o ad albero**, collega tra loro più topologie a stella (FIGURA 32).
La topologia a stella estesa è la più usata nelle moderne reti locali per mantenere tutti i vantaggi della stella in reti che occupano un intero edificio.

FIGURA 32 Topologia a stella estesa (LAN)



TOPOLOGIA A MAGLIA COMPLETA

Questa topologia si usa quando non devono esserci assolutamente interruzioni. Infatti è completamente fault-tolerance, in quanto un guasto a un nodo o a un canale non interrompe il funzionamento della rete, esistendo molti percorsi tra i nodi (FIGURA 33).

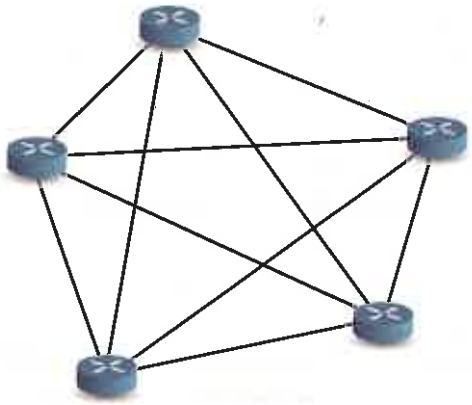


FIGURA 33 Topologia a maglia completa (WAN)

La topologia a maglia è di solito usata nelle reti geografiche per connettere pochi nodi (router), cruciali per la comunicazione a livello nazionale. Infatti l'elevato numero di canali richiesti la rende poco economica all'impiego nelle reti locali o metropolitane.

TOPOLOGIA A MAGLIA PARZIALE

Questa topologia è simile alla maglia completa, ma con un numero inferiore di canali: infatti non tutti i nodi sono connessi con tutti gli altri nodi (FIGURA 34).

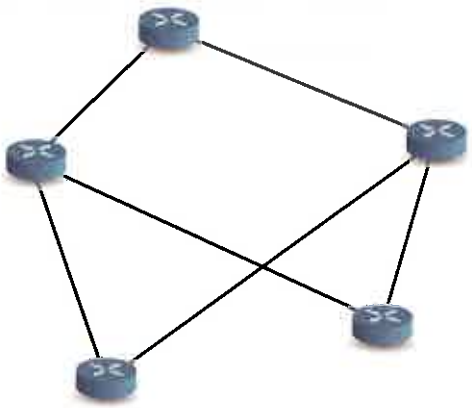


FIGURA 34 Topologia a maglia parziale (WAN)

Mantiene comunque una buona tolleranza ai guasti e ha il vantaggio di lasciare libero il progettista nella scelta del numero di canali da usare.
Questa topologia di rete è la più usata nelle reti geografiche.

FISSA LE CONOSCENZE

- Come si possono classificare le reti in base alla loro estensione?
- Che cosa si intende per topologia di rete?
- Descrivi la topologia di rete a stella. A quali tipi di rete si applica?
- Descrivi la topologia di rete a maglia parziale. A quali tipi di rete si applica?

Case study
Progettare una rete da ufficio