

## Accesso al canale trasmissivo

Un'altro dei compiti del DLL è gestire l'**accesso al canale di comunicazione**. Se il canale è punto-punto bidirezionale, non serve neanche un protocollo. Nel caso di comunicazione broadcast si deve cominciare a pensare come gestire l'accesso al canale. Il DLL in questo caso si divide in due parti: LLC (Logical Link Control) e MAC (Media Access Control). Il livello LLC si occupa della gestione dei protocolli di comunicazione e della gestione degli errori, mentre il livello MAC gestisce l'accesso al mezzo trasmissivo.

### Problemi :

- Collisione: si verifica quando due o più nodi tentano di trasmettere contemporaneamente nello stesso canale di comunicazione, causando un conflitto e la perdita di dati.

**TDMA** (time division multiple acces) : il tempo viene suddiviso in slot temporali di durata fissa e assegnati ai vari nodi per la trasmissione dei dati. Ogni nodo può occupare uno o più slot di tempo, a seconda delle necessità di comunicazione. Il sistema di gestione del canale, stabilisce gli slot temporali e li assegna ai vari nodi. Una volta stabiliti gli slot, essi rimangono assegnati ai rispettivi proprietari e non cambiano a meno che non vengano apportate modifiche al sistema di gestione del canale.

**FDMA** (frequency division multiple acces) : le frequenze vengono suddivise in canali e assegnati ai vari nodi per la trasmissione dei dati. Ogni nodo occupa un canale di frequenza specifico e non lo condivide con altri nodi. In altre parole, ogni nodo utilizza lo slot di frequenza assegnatogli per tutto il tempo, ma solamente il suo slot di frequenza. La suddivisione delle frequenze in canali e l'assegnazione dei canali ai nodi sono stabiliti dal sistema di gestione del canale, che cerca di minimizzare le interferenze tra i nodi.

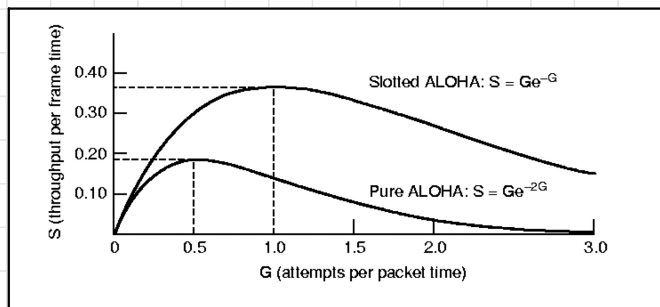
**CDMA** (code division multiple acces) : ogni nodo utilizza un codice unico per codificare i propri dati prima di trasmetterli sul canale, e ogni ricevente applica il codice corrispondente per estrarre i dati originali dal segnale. Per evitare interferenze tra le comunicazioni, i codici devono essere tra loro indipendenti, ovvero non correlati tra di loro.

## ALOHA

La regola è che parla quando vuole, se deve comunicare trasmette, altrimenti no. Si usa un'antenna per trasmettere e una ricevente nelle stazioni periferiche, con un'antenna centrale che fungeva da ripetitore. Quando un ente vuole trasmettere la sua frame può farlo : la frame arrivava all'antenna centrale che la ripete su un determinato canale. In caso di collisione viene trasmesso il segnale rovinato, altrimenti il segnale arriva correttamente a destinazione.

La situazione ideale è data dall'assenza di perdita per qualunque richiesta di traffico che non superi la soglia di traffico massimo. All'atto pratico si hanno delle perdite anche al di sotto di quella soglia in quanto il traffico richiesto ( $G$ ) supera quello massimo consentito ( $S$ ).

**Il canale viene utilizzato al più per il 18.4% delle sue potenzialità quando  $G = 0.5$ , infatti  $S = 0.184$ .**



Non si può evitare la collisione quindi si sfrutta il fatto che statisticamente pochi utenti hanno necessità di iniziare la trasmissione contemporaneamente e quindi si ha bassa probabilità di sovrapposizione di slot. Più si richiede al canale e più si abbassa il throughput ottenibile.

Una macchina poteva rilevare se una frame era stata rovinata prima ancora di ricevere riscontro dai livelli superiori, semplicemente ascoltando il canale di ritorno. Se sul canale di ritorno riesce a sentire la sua frame vuol dire che non è stata rovinata, se sente rumore invece il trasmettitore deve ritrasmettere il messaggio.

Quindi nonostante non fosse definito in maniera esplicita, di fatto c'è un **controllo di collisione**, proprio perché i canali di andata e ritorno erano separati.

## **Carrier sense**

Prima di trasmettere, un dispositivo utilizza il Carrier Sense per ascoltare il canale e rilevare se c'è già una trasmissione in corso. Se rileva una portante di modulazione o un'altra forma di segnale di trasmissione, il dispositivo rimane in silenzio e attende il momento appropriato per trasmettere. Questo aiuta a ridurre le collisioni tra le trasmissioni

## **CSMA 1-persistent**

Un dispositivo tenta di trasmettere immediatamente dopo aver rilevato la fine di una trasmissione precedente. Se più dispositivi utilizzano la stessa strategia e rilevano la fine della trasmissione contemporaneamente, potrebbe verificarsi una collisione. In tal caso, i dispositivi devono attenderne la risoluzione e ritentare in seguito.

## **CSMA p-persistent**

Un dispositivo rileva la fine di una trasmissione precedente, genera un numero casuale e se il numero generato è inferiore o uguale a una certa probabilità  $p$ , il dispositivo inizia la trasmissione. Questa strategia offre la possibilità di ottenere un throughput quasi massimo per un dispositivo fortunato, in quanto il resto del traffico è stato tagliato via dal meccanismo di probabilità.

## **CSMA non-persistent**

Un dispositivo aspetta un periodo di tempo casuale dopo aver rilevato la fine di una trasmissione precedente prima di tentare la trasmissione. Questo aiuta a evitare la sincronizzazione tra i dispositivi e riduce la probabilità di collisioni ripetute. Tuttavia, l'attesa casuale può causare un leggero ritardo nella trasmissione dei dati.

## **Aloha e CSMA non risolvono il problema delle collisioni.**

## **CSMA/CD (carrier sense multiple access with collision detection)**

Si ha un numero di slot pari al numero di stazioni (numerate) che possono parlare contemporaneamente nel mezzo. Si invia un segnale di beacon agli utenti che vogliono comunicare ed essi comunicano la propria volontà di voler comunicare settando un apposito bit e si saprà esattamente chi è interessato a parlare e l'ordine di prenotazione. Terminato il periodo di contesa, tutte le macchine sanno chi vuole parlare, avendo ascoltato il canale, e allora trasmetteranno in ordine. Non garantisce la totale assenza di collisione, ci può ancora essere perché qualcuno potrebbe non rispettare il protocollo.

### **Problemi con CSMA/CD :**

- Se ci sono tante macchine il periodo di contesa diventa inutilmente lungo ed è uno schema utile se tutte le macchine presenti tendono a comunicare. In caso contrario il canale rimane inutilizzato per gran parte del tempo.
- Deve essere noto il numero di macchine presenti nel sistema e non si possono aggiungere/togliere macchine a piacimento. L'aggiunta o rimozione di slot deve essere sincronizzata con le altre macchine.

## CSMA/BA

Gli utenti comunicano utilizzando segnali a due livelli: uno segnale alto per trasmettere il bit 1 e il silenzio per trasmettere il bit 0. Se due macchine dicono contemporaneamente "1", il segnale non va in collisione e il risultato è un bit 1. Se una macchina dice "0" e l'altra dice "1", prevale il bit 1 e lo 0 scompare senza collisioni.

Bit time	
	0 1 2 3
0 0 1 0	0 - - -
0 1 0 0	0 - - -
1 0 0 1	1 0 0 -
1 0 1 0	1 0 1 0
Result	1 0 1 0

Alla fine comunica la macchina con il MAC adress più alto ed inoltre alla fine si ha già nel canale il MAC del mittente e non si spreca lo spazio per slot di contesa. Il problema è dato dal fatto che i MAC address più alti saranno avvantaggiati nella comunicazione, ma è un modo per garantire la priorità a delle macchine privilegiate.

## Token bus

Si da un messaggio (token) ad una macchina e chi ha il token occupa il canale, al termine passa il token alla macchina successiva. Il token è una frame e potrebbe collidere con altre frame o anche perdersi a causa di interferenze. Se si perde il token non finisce tutto. Se dopo un certo tempo tutte notano che il token non sta girando allora tutte sono autorizzate a generare un nuovo token, nella speranza che venga generato un solo token alla volta.

Se ci sono due token si deve prevedere un sottoprotocollo per la rimozione del superfluo. Poi una macchina deve poter entrare/uscire dall'anello. Uscire è facile, ma come si fa per entrare? Un sottoprotocollo deve poter gestire questi casi. Questo approccio evita le collisioni ma è di difficile implementazione.

## Ethernet RFC->802.3 (CSMA/CD)

Il protocollo CSMA/CD è stato sviluppato per gestire i canali di trasmissione condivisi, come quelli utilizzati nell'Ethernet, in cui più dispositivi competono per l'accesso al canale. La combinazione di "carrier sense" e "collision detection" consente di ottimizzare l'efficienza della trasmissione e di gestire efficacemente le collisioni quando si verificano.

Per determinare l'intervallo di attesa prima di un nuovo tentativo, Ethernet utilizza un algoritmo di **backoff esponenziale**. Se un dispositivo non riesce a trasmettere correttamente, ha la possibilità di ritentare la trasmissione fino a un certo numero di tentativi, solitamente 10. Il dispositivo seleziona casualmente un numero nell'intervallo da 0 a un valore massimo specificato, che cresce con ogni tentativo.

Se si verifica una collisione, il dispositivo selezionerà casualmente un tempo di attesa nell'intervallo da 0 a 10 secondi per il secondo tentativo. Nel caso di ulteriori collisioni, il dispositivo selezionerà casualmente un tempo di attesa nell'intervallo da 0 a 20 secondi per il terzo tentativo, e così via. Il valore massimo dell'intervallo di attesa cresce in modo esponenziale, fino a un massimo di 1023. In questo modo ogni dispositivo che ha subito una collisione aspetta un tempo casuale prima di ritentare la trasmissione. Ciò aumenta le possibilità che un dispositivo trovi il canale libero e possa trasmettere senza collisioni.

Codifica Manchester

Ci sono due livelli, alto basso, il bit 1 è identificato da un passaggio alto-basso, mentre il bit 0 da un passaggio basso-alto. Si identificano i bit con la **variazione dello stato logico** e non con il livello perchè è più facile identificare il passaggio da uno stato logico all'altro più che un livello fermo. Si necessita di inviare un **preambolo** per sincronizzare mittente e destinatario, ovvero una sequenza di bit terminata da **11** , dove **11** è un'onda quadra di frequenza 20 MHz, questo permette al destinatario di sincronizzarsi col mittente.

Ethernet 10Mbps

Lo standard Ethernet 802.3 sfrutta la codifica machester

Name	Cable	Max. seg.	Nodes/seg.	Advantages
10Base5	Thick coax	500 m	100	Original cable; now obsolete
10Base2	Thin coax	185 m	30	No hub needed
10Base-T	Twisted pair	100 m	1024	Cheapest system
10Base-F	Fiber optics	2000 m	1024	Best between buildings

- Il primo è il cavo coassiale grosso **10Base5**, molto rigido perché era un pezzo di rame unico non intrecciato, una guaina isolante e una calza metallica di protezione. Ovviamente poi c'è la guaina esterna di protezione totale. La lunghezza massima per un segmento unico era di 500 metri, e potevano essere collegati tra di loro con dei ripetitori / amplificatori di segnale fino a un massimo di 5 segmenti consecutivi, quindi 2.5km.
- Per quanto riguarda **10Base2** è molto più flessibile e per estendere il cavo semplicemente si tagliava (cosa che rovina il segnale) e si inseriva il connettore. Non avere più il cavo unico è un problema e quindi ha segmenti massimi di 185 metri, sempre 5 segmenti massimo fra di loro, quindi arriva a stento a 1km.

**Il cavo coassiale funziona meglio di quello intrecciato ma bisogna inserire un terminatore nel cavo , in assenza del quale la comunicazione non può avvenire .**

- 10Base-T** (doppino) è un cavo a coppie intrecciate, ha 4 coppie di cavi intrecciati, i quali vengono intrecciati a loro volta. Questi cavi permettono comunicazione massimo 100 m con più nodi, questo perché lo schema di collegamento prevedeva un hub centrale, un concentratore, e la connessione era diretta tra scheda di rete e concentratore. Possiamo vederlo anche come un centro a stella.
- L'ultimo schema è **10Base-F**, la fibra ottica (la velocità è sempre la stessa). Permette una velocità di 10Mbit al secondo , prestazioni eccezionali e permette collegamenti in lunga distanza fino a 2km senza ripetitori.

Categorie dei cavi :

Trasmissione a		Cavi	
Category	Data Rate	Signal Frequency	Standard
Cat5	100 Mbps	100 MHz	TIA/EIA
Cat5e	100 Mbps /1 Gbps	100 MHz	TIA/EIA-568-B
Cat6	1Gbps / 10 Gbps	250 MHz	TIA/EIA-568-B
Cat6a	1Gbps / 10 Gbps	500 MHz	ANSI/TIA/EIA-568-B.2-10

## Fast ethernet

Quando ethernet è passato a 100Mbps sono scomparsi i cavi coassiali e sono comparsi i cavi a categoria. La codifica utilizzata è 4B5B.

Più sono regolari e intrecciati le coppie di cavi e più si ha una alta categoria .

Nel caso di cavo categoria 3 (T4) so tutte e 4 le coppie presenti nel cavo, una in andata, una in ritorno, alla velocità di 33 Mbps ciascuna, le ultime due coppie le utilizzo alternativamente (insieme) o in una direzione o nell'altra, in modo tale che ho 3x33Mbps in una direzione e 33 nell'altra. Quindi Fast Ethernet permette di ottenere circa **100Mbps Half Duplex** in una direzione e 33 nell'altra. In alternativa, nel caso di cavo categoria 5 (TX), le altre due coppie sono utilizzate per trasportare corrente continua **POE (power over ethernet)** per iniettare energia elettrica nel dispositivo. Non serve la codifica manchester e nemmeno il cavo coassiale.

Name	Cable	Max. segment	Advantages
100Base-T4	Twisted pair	100 m	Uses category 3 UTP
100Base-TX	Twisted pair	100 m	Full duplex at 100 Mbps
100Base-FX	Fiber optics	2000 m	Full duplex at 100 Mbps; long runs

## Gigabit ethernet

Quando Ethernet passa alla velocità trasmissiva di 1000Mbps i cavi sono solo twisted (doppini) ed in fibra ottica. Cinque passi verso 1000Base-T (Cat5) :

- Rimuovere codifica 4B5B (100 -> 125 Mbps) : già solo rimuovendo questa codifica aumentiamo il throughput, ma ritorna il problema del framing.
- Usare le 4 coppie simultaneamente (125 -> 500 Mbps) : Fast Ethernet ne usa solo 2, una di andata e una di ritorno. Usando tutte e quattro le coppie simultaneamente possiamo quadruplicarlo.
- Trasmissione full duplex (500 Mbps full duplex) : conseguenza dell'uso delle 4 coppie simultanee.
- Usare 5 livelli per baud invece che 3 (MLT-3) (500 Mbps -> 1Gbps full duplex) : un livello viene usato per fare framing, gli altri 4 permettono di trasportare 2 bit al livello. Da 500 passiamo a 1 Gbps. L'unico problema è che il tasso di errore è così alto che ogni frame sarebbe soggetta a errore.
- Usare Forward Error Correction per recuperare 6 dB : correzione errori a destinazione per sistemare il problema.

Name	Cable	Max. segment	Advantages
1000Base-SX	Fiber optics	550 m	Multimode fiber (50, 62.5 microns)
1000Base-LX	Fiber optics	5000 m	Single (10 μ) or multimode (50, 62.5 μ)
1000Base-CX	2 Pairs of STP	25 m	Shielded twisted pair
1000Base-T	4 Pairs of UTP	100 m	Standard category 5 UTP

Tecnologia	Massima lunghezza del link	Codifica	Topologia del mezzo		Bit rate (bps)
10Base5	500 m	Manchester	bus	50-ohm coax	10 M
10Base2	185 m	Manchester	bus	50-ohm coax	10 M
10BaseT	100 m	Manchester	star	2 pair UTP cat. 3,4,5	10
100BaseFL	2000 m	Manchester	star	Multi-mode fiber*	10 M
100BaseT2	100 m	PAM 5x5	star	2 pairs UTP cat. 3,4,5	100 M
100BaseT4	100 m	8B/6T	star	4 pairs UTP cat. 3,4,5	100 M
100BaseTX	100 m	4B/5B with MLT-3	star	2 pairs UTP cat. 5	100 M
100BaseFX	412/2000 m	4B/5B with NRZI	star	Multi-mode fiber*	100 M
1000BaseT	100 m	PAM 5x5	star	4 pairs UTP Cat 5	1000 M
1000BaseSX	275 m	8B/10B	star	Multi-mode fiber†	1000 M
1000BaseLX	316/550 m	8B/10B	star	Multi-modeFiber‡	1000 M
1000BaseCX	25 m	8B/10B	star	Twinax	1000 M

### Codifica di Travis :

Si ha un automa a stati finiti che predefinisce un percorso univoco per ogni sequenza possibile di distanza 2 e si ricostruisce facilmente il messaggio di partenza basandosi sulla probabilità. Questo meccanismo è in grado di rilevare e correggere anche errori doppi per ogni byte (un tasso di errore enorme di un bit ogni 4).

A questa codifica segue un controllo CRC per garantire la correttezza dei dati. Questi metodi combinati forniscono una maggiore affidabilità nella trasmissione dei dati e la possibilità di rilevare e correggere una grosso tasso di errori.