

Livello Data Link

•
Davide Quaglia

Scopi

- Comunicazione affidabile ed efficiente tra 2 macchine sullo stesso mezzo fisico
 - Cavo
 - Circuito virtuale telefonico
 - Wireless
- Si assume:
 - Esista l'entità *bit* (dal livello fisico)
 - I bit, se ricevuti, arrivano nell'ordine in cui sono stati inviati

Funzionalità

- Rilevamento degli errori ed eventuale ritrasmissione di dati errati
- Politiche di utilizzo del canale in caso di più di 2 macchine sul mezzo fisico
 - Indirizzamento delle macchine
 - Arbitraggio del canale
- *Framing* = raggruppamento di bit in PDU di livello 2

Servizi offerti al livello Network

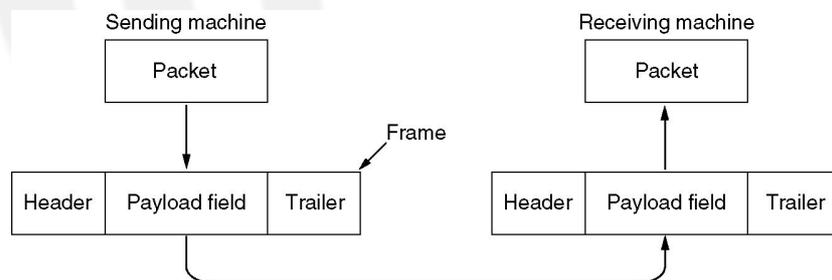
- Servizio di trasmissione non connesso e non confermato (es. Ethernet/802.3)
- Servizio non connesso ma confermato (con acknowledge) (es. WiFi/802.11)
- Servizio connesso (es. PPP, 802.16)

- NOTA: i servizi connessi sono anche affidabili e con controllo di flusso perché numerano le PDU e ne confermano la ricezione (acknowledge)

Framing

- Migliora l'utilizzo del canale in caso di più di due stazioni
- Gestione errori
 - Rilevazione con codici a blocco (es. CRC)
 - Recupero mediante ritrasmissione di PDU
- Problema: definizione di inizio/fine di PDU

Framing



Inizio/fine frame

- Occorre utilizzare simboli che non possano comparire nel corpo del frame
 - Violazioni della codifica di bit
 - Particolari configurazioni possono facilitare la sincronizzazione di bit tra TX e RX
 - Utilizzo di una sequenza particolare di bit (FLAG)
 - Bit stuffing/de-stuffing per evitare emulazione del FLAG nella PDU
 - Forzare un intervallo (inter-packet gap) minimo tra 2 PDU consecutive

Bit stuffing/de-stuffing

- Il protocollo HDLC e il progenitore PPP usano il byte 01111110 come FLAG a inizio e fine frame
- I bit del frame originale vengono sottoposti a *stuffing*
 - Dopo cinque “1” viene inserito automaticamente uno “0”
- Al ricevitore il livello Data Link fa il processo inverso (de-stuffing)

Bit stuffing: esempio

(a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

(b) 0 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0

↑
Stuffed bits

(c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

- a) Dati originali.
- b) Dati trasmessi sul canale.
- c) Dati nel ricevitore dopo de-stuffing.

Rilevazione di errori

- Certi valori di bit vengono interpretati erroneamente al RX
 - Gli errori spesso compaiono in sequenze dette *burst* (raffiche)
- Distanza di Hamming
- Occorre aggiungere *bit ridondanti* al messaggio
 - m bit di messaggio
 - r bit di codice per la rilevazione dell'errore
 - $n=m+r$ bit trasmessi sul canale
 - Velocità del codice (rate) = m/n
- Esempi
 - Bit di parità
 - Checksum
 - Circular Redundancy Check (CRC)

Bit di parità

- Al trasmettitore viene aggiunto un bit in coda al messaggio
 - “1” se il numero di “1” nel messaggio è pari
 - “0” se il numero di “1” nel messaggio è dispari
- Al ricevitore se il numero di “1” è pari significa che c’è stato errore
 - Rileva solo errori che riguardano un numero dispari di bit
 - Errori su un numero pari di bit passano inosservati

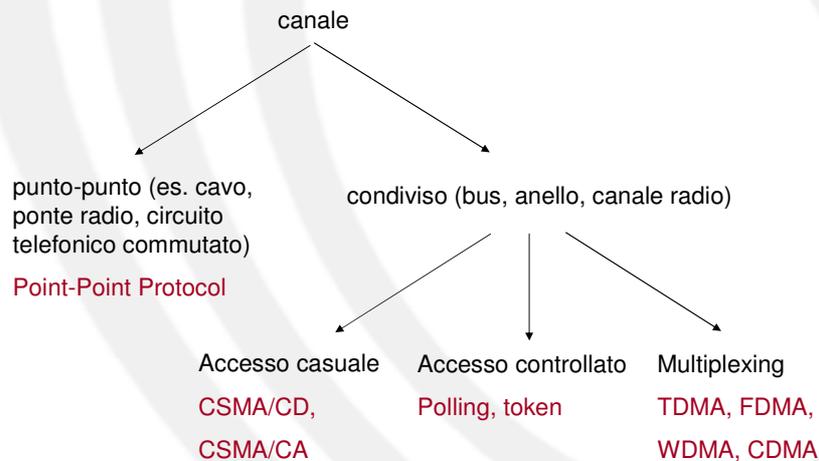
Check sum

- Estensione del metodo del bit di parità
- Divisione del messaggio in parole di r bit
- Somma su r bit di tutte le parole ignorando overflow
- La somma viene accodata al messaggio
- La somma viene ricalcolata al ricevitore
- Errori multipli che non cambiano la somma non vengono rilevati

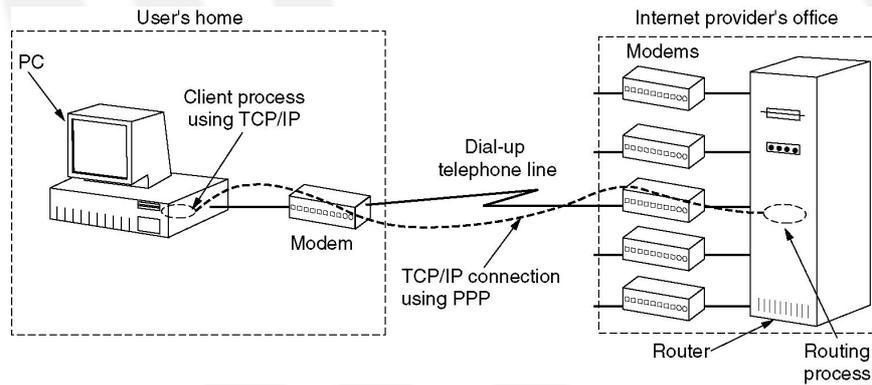
Circular Redundancy Check (CRC)

- Il messaggio viene interpretato come il vettore dei coefficienti di un polinomio $M(x)$ di grado $m-1$
- Sia $R(x)$ il resto della divisione tra polinomi $x^r M(x)/G(x)$ dove $G(x)$ è detto *polinomio generatore*
- Il polinomio $x^r M(x) - R(x)$ è sempre divisibile per $G(x)$ e viene trasmesso sul canale ($m+r$ bit)
- Al ricevitore se la sequenza di bit è divisibile per $G(x)$ allora si assume corretta

Metodi di accesso al canale



Point-to-point protocol (PPP)



PPP: struttura del frame

- Mutua i frame di tipo *un-numbered information* di HDLC (campo *control*=0x03)
- Indirizzo destinazione inutile e messo a costante (topologia punto-punto)

Bytes	1	1	1	1 or 2	Variable	2 or 4	1
	Flag 01111110	Address 11111111	Control 0000011	Protocol	Payload	Checksum	Flag 01111110

PPP: diagramma stati di una connessione

