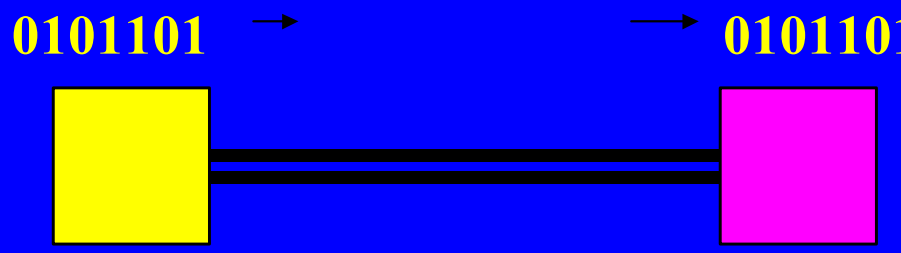
5. Siirtoyhteyskerros

linkkikerros (Data Link Layer)

- yhtenäinen linkki solmusta solmuun
  - bitit sisään => bitit ulos
- ongelmia:
  - siirtovirheet
    - havaitseminen
    - korjaaminen
  - solmun kapasiteetti
    - vuonvalvonta
  - yhteisen kanavan käyttö

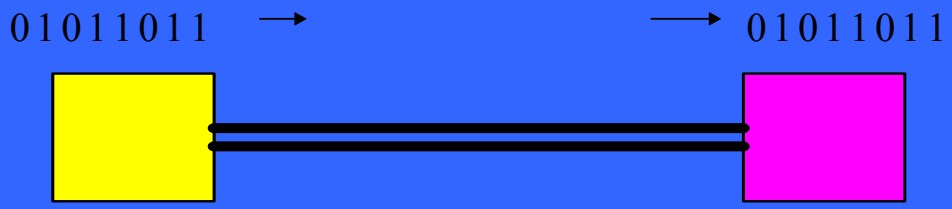


5.1. Kaksipisteyhteydet

Virhevalvonta

* virheiden havaitseminen
* virheiden korjaus

Vuonvalvonta



Bittivirta <=>kehyksiä

- tavoite
  - bittivirheiden hallinta
    - muuttuu
    - katoaa
    - monistuu
- bittivirta kehyksinä
- kehys tarkistettavissa
  - tarkistustietoa

Kehysten kuljetus

- tavoite
  - kaikki kehykset
  - kukin kehys virheettömästi
  - lähetysjärjetyksessä
- vastaanottaja kertoo lähettäjälle
  - ACK: kehys vastaanotettu ok
    - tietty kehys
    - kaikki kehykset tähän asti
  - NAK: kehyksessä vikaa => lähetettävä uudelleen
  - Saako lähettää lisää vai pitääkö keskeyttää
    - vuonvalvonta

Virheet

- Kahdenlaisia virheitä:
  - yhden bitin virheet
  - usean peräkkäisen bitin vääristyminen (burst error)
- Virheiden esiintymistiheys
  - BER (bit error rate)
  - mitä suurempi BER, sitä lyhyempiä kehyksiä kannattaa käyttää

Missä virhe hoidetaan?

- kuittaava linkkikerros havaitsee virheet ja korjaa ne
- yhteydetön, kuittaamaton & virhe

  => kuljetuskerros havaitsee ja korjaa
- ja jos ei, niin sovelluskerros havaitsee ja korjaa
- ja jos ei, niin asiakas havaitsee ja korjaa

Virheiden havaitseminen ja korjaaminen

Virheiden takia dataan lisäinformaatiota:

- **virheen korjaamiseksi** (error-correcting code, forward error correction (FEC))
  - lisäinformaatiota niin paljon, että vastaanottaja sekä havaitsee että kykenee itse korjaamaan virheen
- **virheen havaitsemiseksi** (error-detecting code, feedback/backward error control)
  - lisäinformaatiota, jotta vastaanottaja havaitsee virheen tapahtuneen => korjauksena **uudelleenlähetys**

Virheen korjaus/havaitseminen

- virheen korjaava koodaus
  - kallis koko ajan
    - paljon lisäinformaatiota
  - rajoitettu korjauskyky
    - esim. kokonaan kodonnut kehys
- virheen havaitseva koodaus
  - virheen sattuessa kallis
    - uudelleen lähettäminen maksaa
    - uudelleen lähettäminen on hidasta

Virheen korjaus

- Käytetään esim.
  - CD- ja DVD-levyissä, digitaalitelevisiossa
  - nopeissa modeemeissa, kannettavissa puhelimissa
  - satellittiyhteyksissä, avaruusluotaimissa
- Esimerkkejä
  - Hamming-pariteettitarkistus (Tito-kurssilla)
    - pystyy korjaamaan yhden virheellisen bitin
    - virheryöpyn, jos se jaetaan yhden bitin virheiksi
  - Reed-Solomon -koodit
    - lohkokoodeja , jotka pystyvät korjaamaan virheryöppyjä

Virheen havaitseminen

- **Pariteettibitti**
  - parillinen pariteetti
  - pariton pariteetti
- **horisontaaliset ja vertikaaliset pariteetit**
- **Internet tarkistussumma**
- **CRC** (Cyclic redundancy code (tai check))
  - **yleisesti käytetty virheen paljastusmenetelmä**
  - **perustuu polynomien aritmetiikkaan (modulo2-aritmetiikkaan, XOR)**
  - **useita tarkistusbittejä => havaitaan usean bittivirheen ryöppy**

Pariteetti

- esimerkki yksinkertaisesta virheen havaitsevasta koodista
- jokaiseen merkkiin lisätään yksi ylimääräinen ns. pariteettibitti
  - lisäyksen jälkeen kaikissa merkeissä on parillinen (tai jos niin sovitaan pariton) määrä ykkösiä
- paljastaa kaikki yhden bitin virheet
  - kehyksen pituudesta riippumatta
- ei paljasta kahden bitin virheitä

Pariteettibitin käyttö

- erityisesti asynkronisessa tiedonsiirrossa merkkejä siirrettäessä
- käytännössä paljastaa noin puolet virheellisistä bittijonoista
  - esim. modeemeissa syntyy useita virheitä
  - linjahäiriöt aiheuttavat usein pitkiä virheryöppyjä

Horisontaaliset ja vertikaaliset pariteetit

- järjestetään bittijono kaksiulotteiseen taulukkoon
- lasketaan pariteetti jokaiselle vaaka- ja pystyriville

1001010	1
0111010	0
1110001	0
1000111	0
0011001	1
1011111	0

horisontaaliset pariteetit

taulukon pariteetti

vertikaaliset

Virheiden havaitseminen

- Ei löydä lyhyitä virheryöppyjä, joissa neljä bittiä vaihtuu sopivasti

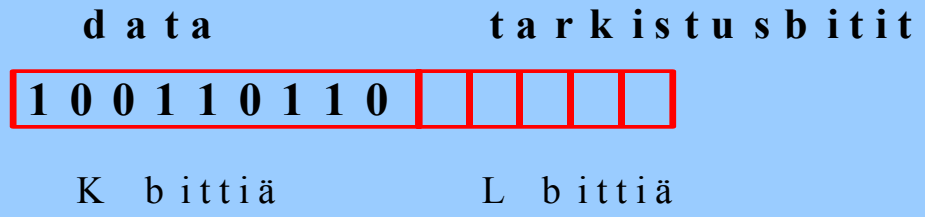
1 0 0 1 0 1 0
0 1 1 1 0 1 0
1 1 1 0 0 0 1
1 0 0 0 1 1 1

Internetin tarkistussumma

- lasketaan 16-bittisten sanojen yhden komplementit yhteen
- otetaan summasta yhden komplementti
- käytetään Intenet-protokollissa
  - UDP- ja TCP -protokollissa
- monia virhekombinaatioita jää havaitsematta
- riittävän hyvä, jos virheitä vähän

CRC:n perusidea

- **tarkistusavain (virittäjä, virittäjäpolynomi)**
  - **bittejä yksi enemmän kuin tarkistusbittejä**
  - **lähettäjä ja vastaanottaja tuntevat**
- **lähettäjä**
  - **laskee lähetettävälle datalle tarkistusavaimen avulla tarkistusbitit ja liittää ne kehykseen**
- **vastaanottaja**
  - **tarkistaa, onko koko saapunut kehys (data + tarkistusbitit) pysynyt muuttumattomana**



Esimerkki: data = **101110**, virittäjä = **1001**,

(polynomina **X**3 +1**), tarkistusbittejä 3

Lähetettävä data = 101110??? tarkistusbitit

```
          101011
      ------------
1001 | 101110000
       1001
       ----
        1010
        1001
        ----
         1100
         1001
         ----
          1010
          1001
          ----
          0011 = tarkistusbitit
```

Modulo 2-aritmetiikka:

1+1 = 0 (XOR)

Lähetetään: **101110 011**

Vastaanottaja:jakaa saamansa kehyksen virittäjällä. Kehys on ok, jos jakojäännös on 0!

Standardoituja virittäjäpolynomeja

- CRC-12 = x**12 + x**11 + x**3 + x**2 + x + 1
- CRC-16 = x**16 + x**15 + x**2 +1
- CRC-32=x**32+ x**26 + x**23+…+ x**4 + x**2 + x+1

CRC: n virheiden havaitsemiskyky

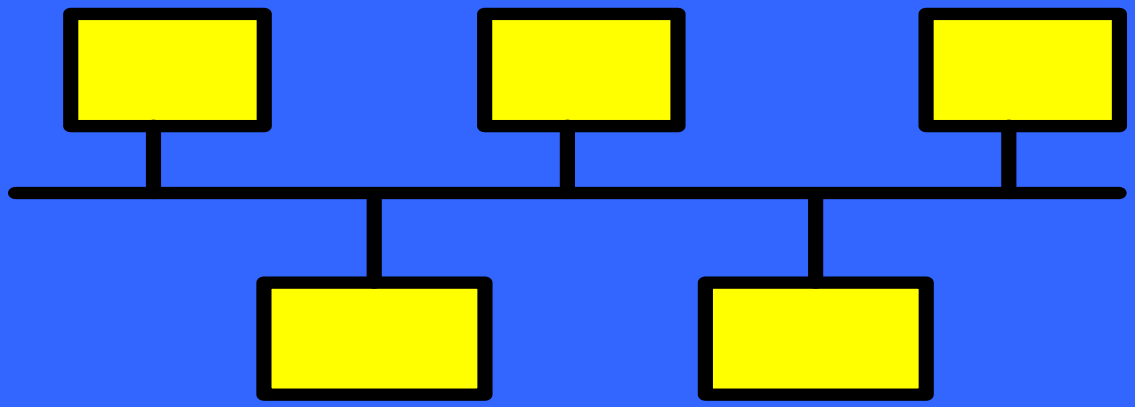
- kaikki virheryöpyt, joiden pituus < tai = kuin virittäjän
- useimmat virheryöpyt, joiden pituus on suurempi
  - CRC-32: P{ryöppy > 33 havaitaan} = 0.9999999998
  - Huom
    - » Arvioinneissa lähtökohtana ollut täysin satunnainen bittien jakautuminen, mutta todellisuudessa näin ei ole!
    - » Joten havaitsemattomien virheiden määrä on arvioitua suurempi.

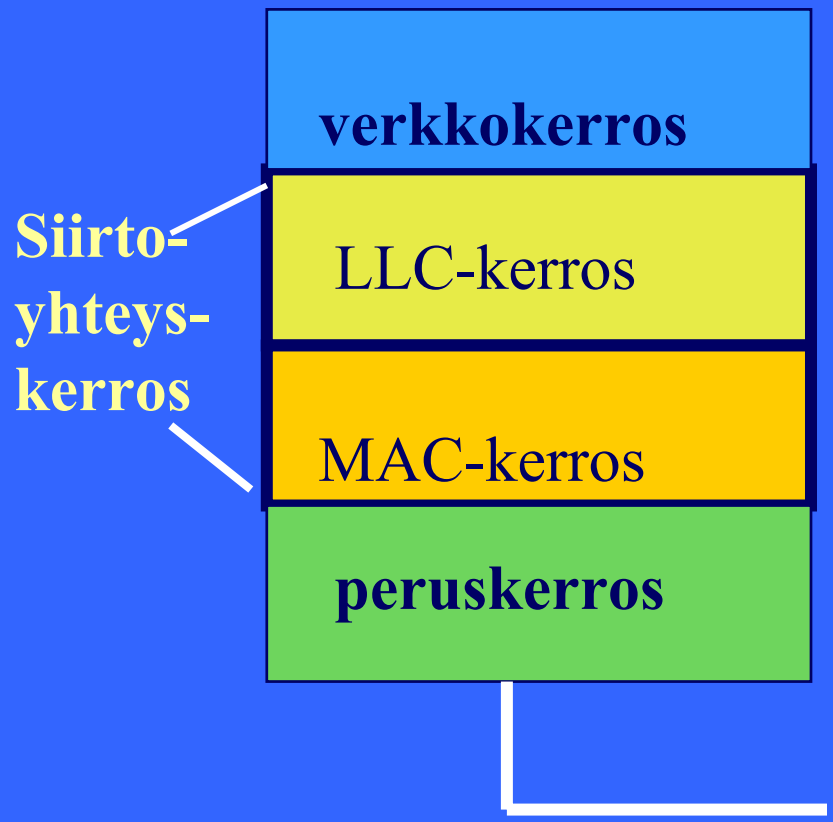
Vuonvalvonta

- Liukuva ikkuna
  - ikkunan koko rajoittaa lähettämistä
    - » jos kehyksen numero ei ole ikkunassa, sitä ei oteta vastaan
  - kuittaus siirtää ikkunaa eteenpäin
- stop-sanoma
  - Receive not ready

5.3. Yhteiskäyttöinen kanava

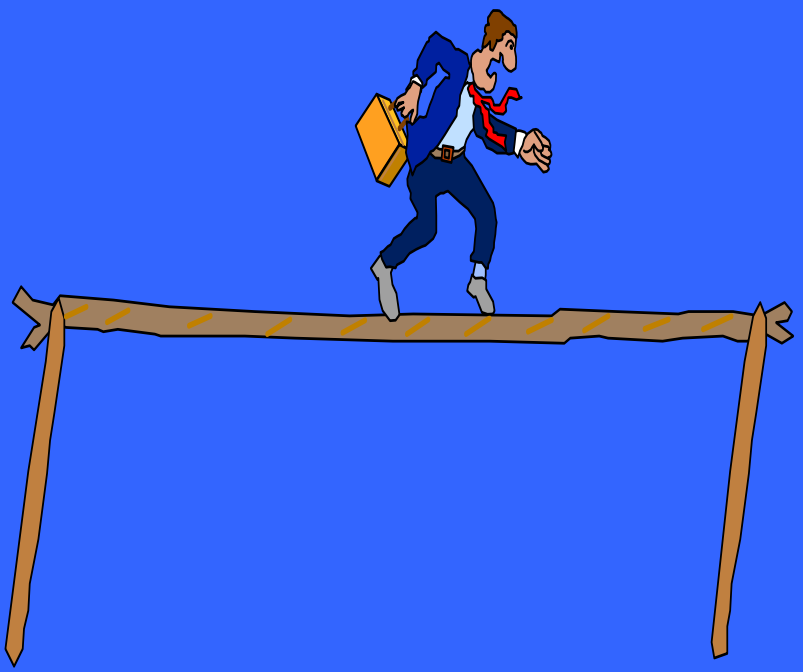
- **yleislähetys (broadcast)**
  - » **multiaccess channel**
  - » **random access channel**
  - – **LAN (Ethernet)**
  - – **langaton**
- **ongelma: käyttövuoron ‘jakelu’**





MAC = Medium Access Control

LLC = Logical Link Control



Vain yksi kerrallaan!

Eri yhteiskäyttötapoja on hyvin paljon:

- **kilpailu** Aloha,CSMA, **CSMA/CD**
  - ‘se ottaa kun ehtii’
- vuorotellen: pollaus, vuoromerkki
  - ‘sinä ensin ja sitten on minun vuoroni’
- varaus: vuorot varataan etukäteen
  - varaukseen käytetään usein kilpailua
- kanava jaetaan: TDMA, FDMA, **CDMA**
  - ‘käytä sinä tätä puolta ja minä tätä toista’

Törmäys

- yksi yhteinen kanava lähettäjille
  - lähetys onnistuu vain, jos yksi lähettää
- Jos useampi kuin yksi lähettää, syntyy **yhteentörmäys** (collision)
  - kaikki törmänneet sanomat tuhoutuvat ja ne on lähetettävä uudelleen
    - vaikka törmäisivät vain yhden bitin verran
  - **kaikkien havaittavissa**
    - LAN: törmäyssignaali
    - satelliittikanava: kuuntelee oman lähetyksensä
    - WLAN: ilmoitus vastaanottajalta

Aika

- **jatkuva aika**
  - lähetykset voivat alkaa milloin vain
  - ei mitään synkronointi, ei yhteistä aikaa
- **viipaloitu aika** (slotted time)
  - aika lokeroitu aikaviipaleiksi
  - lähetys voi alkaa vain aikaviipaleen alussa
  - aikaviipaleessa
    - ei kukaan lähetä => hukkaan
    - yksi lähetys => ok
    - useita lähetyksiä => törmäys
  - vähentää törmäyksiin (=hukkaan) menevää aikaa
    - törmäykset täydellisiä