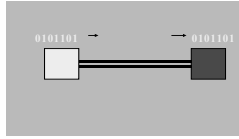


## 5. Siirtoyhteyskerros

linkkikerros (Data Link Layer)

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



2/18/2004

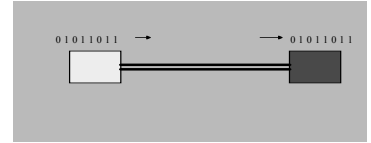
1

## Kaksipisteyhteydet

### Virhevalvonta

- \* virheiden havaitseminen
- \* virheiden korjaus

### Vuonvalvonta



2/18/2004

2

## Bittivirta <=>kehysksiä

- tavoite
  - bittivirheiden hallinta
  - muuttuu
  - katoaa
  - monistuu
- bittivirta jaetaan kehyksiksi (frame)
  - Mistä tiedetään kehysrajat bittivirrassa?
- => kehys tarkistettavissa
  - lisäämällä tarkistustietoa

2/18/2004

3

## 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

2/18/2004

4

## Virheet

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

2/18/2004

5

## 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

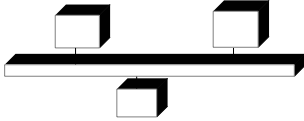
2/18/2004

6

## Yhteiskäyttöinen siirtomedia

### Yleislähetys (broadcast)

- Radiolähetykset, satelliittilähetykset, lähiverkot
  - lähetys onnistuu, jos kullakin hetkellä vain yksi lähettää;
  - usea samanaikainen lähetys => lähetykset tuhoutuvat
- **datan lähetystä täytyy jotenkin säännellä => säännöt määräävä protokolla (MAC)**



2/18/2004

7

## Linkkikerrokset tehtäviä:

- kehystys, jotta virheen havaitseminen mahdollista
- virheiden havaitseminen, jotta virhe huomataan
- virheiden korjaaminen, jotta saadaan oikea tieto
- luotettava tiedonsiirto kuittauksilla
- vuonvalvonta, jotta ei tuhota jo lähetettyjä kehyksiä
- yleislähetyslinkkien lähetysvuorojen sääntely, jotta lähetys ylipäänsä onnistuisi

2/18/2004

8

## 5.2. 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
    - viallisen kehyksen poisheittäminen

2/18/2004

9

## Virheen korjaus/havaitseminen

- **virheen korjaava koodaus**
  - kallis koko ajan
    - paljon lisäinformaatiota
  - rajoitettu korjauskyky
    - esim. kokonaan kodonnut kehys
- **virheen havaitseva koodaus**
  - virheen sattua kallis
    - uudelleen lähettäminen maksaa
      - » siirtokapasiteetti, käsittely
    - uudelleen lähettäminen on hidasta
      - » Aikaa kuluu ennenkuin tulee kuittaus ja saadaan lähetettyä uudestaan

2/18/2004

10

## Virheen korjaus

- Käytetään esim.
  - CD- ja DVD-levyissä, digitaalitelevisiossa
  - nopeissa modeemeissa (esim. ADSL), langattomissa yhteyksissä (kännykkä)
  - satelliittiyhteyksissä, avaruusluotaimissa
- Esimerkkejä
  - **Hamming-pariteettitarkistus** (Tito-kurssilla)
    - » pystyy korjaamaan yhden virheellisen bitin
    - » virheryöpy, jos se jaetaan yhden bitin virheiksi
  - Reed-Solomon -koodit
    - » lohkokoodia, jotka pystyvät korjaamaan virheryöppyjä

2/18/2004

11

## Virheen havaitseminen

- **Pariteettibitti**
  - parillinen pariteetti
  - pariton pariteetti
- **Horizontaaliset 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 ryöpy

2/18/2004

12

## 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

1100111 1

- Pariton määrä bittivirheitä havaitaan
  - kehyksen pituudesta riippumatta
- Mutta ei parillista määrää bittivirheitä

1100011 1

0100011 1

2/18/2004

13

## Pariteettibitin käyttö

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

2/18/2004

14

## Horizontaaliset ja vertikaaliset pariteetit

- järjestetään bittijono kaksikulotteiseen taulukkoon
- lasketaan pariteetti jokaiselle vaaka- ja pystyriivoille

1001010		1	
0111010		0	
1110001		0	horizontaaliset
1000111		0	pariteetit
0011001		1	
1011111		0	taulukon pariteetti
			vertikaaliset
			pariteetit

2/18/2004

15

## Virheiden havaitseminen

- Ei löydy 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

2/18/2004

16

## Internetin tarkistussumma

- lasketaan paketin tai sen otsakkeen 16-bittiset sanat yhteen käyttäen yhden komplementin yhteenlaskua
  - 'ylivuotaneet' muistibitit lisätään summaan
- otetaan summasta yhden komplementti
  - Summan ykköset nolliksi and nollat ykkösiksi
- käytetään Internet-protokollissa
  - UDP- ja TCP -protokollissa
- monia virhekombinaatioita jää havaitsematta
  - riittävän hyvä, jos virheitä vähän

2/18/2004

17

## 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 kehyk (data + tarkistusbitit) pysynyt muuttumattomana

data = D tarkistusbitit = R

100110110 111

d bittia r bittia

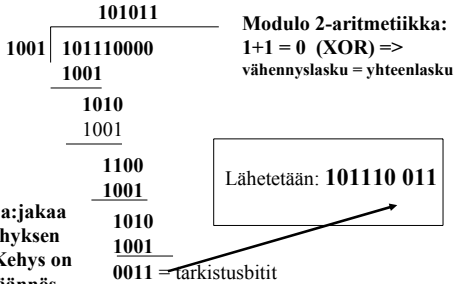
2/18/2004

18

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

( polynomina  $X^{**3} + 1$ ), tarkistusbittejä 3

Lähetettävä data = 101110??? tarkistusbitit



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

## Kuinka näin?

- $D * 2^{**r}$  lisätään r nollaa datan perään = kerrotaan  $2^{**r}$  :llä
- $D * 2^{**r} \oplus R = n G$  data  $\oplus$  tarkistusbitit on tasan jaollinen virittäjällä G ( $\oplus = \text{XOR}$ )
- $(D * 2^{**r} \oplus R) \oplus R = n G \oplus R$  lisätään ( $\oplus$ ) kummallekin puolelle
- $D * 2^{**r} = n G \oplus R$  koska  $R \oplus R = 0$
- $R =$  jakojäännös, kun jaetaan  $D * 2^{**r} / G$

2/18/2004

20

## 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 =  $x^{**32} + x^{**26} + x^{**23} + \dots + x^{**4} + x^{**2} + x + 1$

### CRC: n virheiden havaitsemiskyky

- kaikki virheryöpyt, joiden pituus < tai = kuin virittäjän
- useimmat virheryöpyt, joiden pituus on suurempi
  - CRC-32: P{ryöppy > 33 havaitaan} = 0.999999999
- Huom
  - Arvioinneissa lähtökohtana ollut täysin satunnainen bittien jakautuminen, mutta todellisuudessa näin ei ole!
  - Joten havaitsemattomien virheiden määrä on arvioitua suurempi.

2/18/2004

21

## Vuonvalvonta

### ■ Liukuva ikkuna

- ikkunan koko rajoittaa lähettämistä
  - jos kehyksen numero ei ole ikkunassa, sitä ei oteta vastaan
- kuittaus siirtää ikkunaa eteenpäin

### ■ stop-sanoma

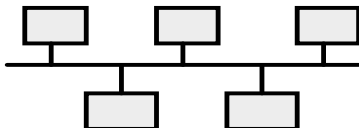
- Receive not ready

2/18/2004

22

## 5.3. Yhteiskäyttöinen kanava

- yleislähetys (broadcast)
  - multiaccess channel
  - random access channel
- LAN (Ethernet)
- langaton
- ongelma: käyttövuoron 'jakelu'



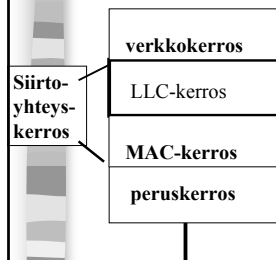
2/18/2004

23

## Linkkikerroksen alikerrokset

MAC = Medium Access Control

LLC = Logical Link Control



Siirto-yhteys-kerros



2/18/2004

24

## 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

2/18/2004

25

## Eri yhteiskäyttötapoja on hyvin paljon:

- **kilpailu** Aloha, CSMA, **CSMA/CD**
  - 'se ottaa, joka ehtii'
- **vuorotellen**: pollaus, vuoromerkki, varaus
  - 'Minä ensin, sitten sinä'
  - Vuorot jaetaan jollakin tavalla
  - Varauksessa: **varaukseen käytetään usein kilpailua**
- **kanava jaetaan**: TDMA, FDMA, **CDMA**
  - 'käytä sinä tätä puolta ja minä tätä toista'

2/18/2004

26

## Hyviä ominaisuuksia MAC-protokollalle:

- **Pieni yleisrasite**:
  - kun yksi lähettää se pystyy hyödyntämään koko kanavan kapasiteetin R bps
- **Tasapuolisuus**:
  - kun M lähettäjää, kukin saa keskimäärin saman osuuden linjan kapasiteetista (R/M bps)
- **Toimintavarmuus**:
  - hajautettu vuoroistapiminen
- **Kustannustehokas**:
  - Yksinkertainen ja halpa toteuttaa

2/18/2004

27

## Yleislähetysprotokollia

### Esimerkkejä:

- **CDMA**
  - radiolinjoilla käytetty koodinjakoon perustuva protokolla
  - TDMA, FDMA
    - Ei törmäyksiä, mutta tuhlaavat kapasiteettia;
    - Jokainen saa käyttöönsä aina vain R/M bps, vaikka olisi ainoa lähettäjä!
- **CSMA/CD** ()
  - mm. Ethernet-verkossa käytetty kilpailuprotokolla
  - Aloha ja CSMA edeltäjiä

2/18/2004

28

## Kanavan jakoprotokollat

- **TDMA**
  - aikajako
    - asemalla oma aikaviipale
- **FDMA**
  - taajuusjako
    - asemalla oma taajuusalue
- **CDMA**
  - koodijako
    - asemalla oma koodi
    - asemat voivat lähettää yhtäaikaan!

2/18/2004

29

## CDMA (Code Division Multiple Access)

- **yksi kanava**
  - usea samanaikainen lähetys
  - kukin koko kanavan taajuudella!
- yhden bitin lähetyssaika jaetaan pienempiin osiin (aikasiruihin)
  - 64 tai 128 sirua bittiä kohden
- kullakin asemalla oma 'sirukuvio' 1-bitin lähetykseen
  - 0-bitti on tämän yhden komplementti

2/18/2004

30

## Esimerkiksi:

- aseman A 1-bitti: 00011011  
0-bitti: 11100100
- aseman B 1-bitti: 00101110  
0-bitti: 11010001
- aseman C 1-bitti: 01011100  
0-bitti: 10100011
- aseman D 1-bitti: 01000010  
0-bitti: 10111101

Ps. Oikeasti käytetään 64 tai 128 sirua

2/18/2004

31

## Laskemisen helpottamiseksi

- koodataan sirut 0 ja 1 seuraavasti:
  - 1 = 1
  - 0 = -1
- aseman A 1-bitti: 00011011 = -1 -1 -1 1 1 -1 1 1  
0-bitti: 11100100 = 1 1 1 -1 -1 1 -1 -1
- aseman B 1-bitti: 00101110 = -1 -1 1 -1 1 1 1 -1  
0-bitti: 11010001 = 1 1 -1 1 -1 -1 -1 1
- aseman C 1-bitti: 01011100 = -1 1 -1 1 1 1 -1 -1  
0-bitti: 10100011
- aseman D 1-bitti: 01000010 = -1 1 -1 -1 -1 -1 1 -1  
0-bitti: 10111101

2/18/2004

32

Kaikki bittikuviot **parittain ortogonaalisia**:

- $A \bullet B = 0 = 1/m \sum A_i B_i$  (sisätulo)
- $A \bullet A = 1$
- $-A \bullet A = -1$
- $\Rightarrow$  yhteissignaalista löydetään eri asemien omat lähetykset!

A:n 1-bitti: 00011011 = -1 -1 -1 1 1 -1 1 1  
B:n 1-bitti: 00101110 = -1 -1 1 -1 1 1 1 -1

$$\begin{aligned} A \bullet B &= 1+1+-1+-1+-1+-1+-1 \\ &= 0 \Rightarrow \text{keskenään ortogonaalisia} \end{aligned}$$

2/18/2004

33

- kukin asema lähettää omat 1-bittinsä ja 0-bittinsä
- kun moni lähettää samanaikaisesti tuloksena on **yhteissignaali S**.
  - lähetettyjen signaalien 'summa'
- aseman datan 'purkaminen' yhteissignaalista
  - A = aseman oma bittikuvio
  - $S \bullet A$  tuottaa aseman lähettämän bitin
    - $\gg$  kerrottuna bitin aikasirujen lukumäärällä

2/18/2004

34

## Esimerkki

- merkintä 1 = 1, 0 = -1,
- helpompi laskea yhteen

- $S = (-2 -2 0 -2 0 -2 4 0)$
- $C = (-1 1 -1 1 1 1 -1 -1)$
- $S \bullet C = (2 -2 0 -2 0 -2 -4 0)$   
 $= -8 \Rightarrow -1$
- eli **C lähetti 0-bitin**

2/18/2004

35

## Esimerkki jatkuu:

Mitä B lähetti?

- $S = (-2 -2 0 -2 0 -2 4 0)$
- $B = (-1 -1 1 -1 1 1 1 -1)$
- $S \bullet B = (2 2 0 2 0 -2 4 0)$   
 $= 8 \Rightarrow 1$
- eli **B lähetti 1-bitin**

2/18/2004

36

## Esimerkki jatkuu

Entä mitä A lähetti?

$$\blacksquare S = (-2 \ -2 \ 0 \ -2 \ 0 \ -2 \ 4 \ 0)$$

$$\blacksquare A = (-1 \ -1 \ -1 \ 1 \ 1 \ -1 \ 1 \ 1)$$

$$\blacksquare S \bullet A = (2 \ 2 \ 0 \ -2 \ 0 \ 2 \ 4 \ 0) \\ = 8 \Rightarrow 1$$

eli A lähetti 1-bitin

■ Lähettikö myös D jotain?

2/18/2004

37

## Käytännössä CDMA on vaativa toteuttaa

- 64 tai 128 bitin ortogonaalisia koodeja
- edellyttää signaalien voimakkuuksien vertailua ja yhteenlaskua => signaalien heikkeneminen eri etäisyyksillä otettava huomioon
- tarkat ajoitukset
- tunnettava lähettäjien sirukoodit sirukoodit

2/18/2004

38

## Kilpailuprotokollia (Random Access Protocols)

- Aina kun asema haluaa lähettää, se pyrkii lähettämään
  - Se yleensä ensin kuuntelee, onko joku muu asema jo lähettämässä. Jos mikään asema ei läheta, niin aloitetaan lähettäminen.
- Jos tulee törmäys, niin odotetaan satunnainen aika ennen uudelleenlähettämistä

Esimerkkejä:

- Aloha
- CSMA
- CSMA/CD
  - mm. Ethernet-verkossa käytetty kilpailuprotokolla

2/18/2004

39

## Lähetyiskanavan kuuntelu (carrier sense)

- käynnissä olevan lähetyksen havaitseminen
  - asema tutkii, onko kanava jo käytössä
    - ennen lähetystä tutkitaan, onko joku muu lähettämässä
    - jos on, ei lähetetä
  - yleensä lähiverkot (CSMA)
  - asema ei tutki kanavan käyttöä
    - asema lähettää aina kun haluaa
    - lähettämisen jälkeen havaitaan onnistuiko
    - esim. satelliittilähetykset

2/18/2004

40

## Kanavan kuuntelu

- ei aina paljasta jo alkanutta lähetystä
  - etenemisviipeen takia
- tai ole järkevää
  - esim. satelliittikanavan kuuntelu ei paljasta sitä, onko joku toinen maa-asema jo aloittanut lähetyksen
  - langattomassa lähiverkossa lähettäjän ympäristön kuuntelu ei kerro sitä, onko vastaanottaja saamassa sanomia muualta

2/18/2004

41

## ALOHA

- Hawaiilla, 70-luvulla radiotietä varten
- puhdas ALOHA (pure Aloha):
  - asema lähettää aina heti, kun sillä on lähetettävää;
    - ei mitään kuuntelua ennen lähetystä
  - ja samalla kuuntelee, onnistuiko lähetykset
    - lähiverkossa törmäys havaitaan 'heti', sillä siirtoviive pieni
    - toisin kuin satelliittilla!
  - jos törmäys, niin lähettäjä odottaa satunnaisen ajan ja yrittää uudelleen
  - maksimaalinen tehokkuus ~18%

2/18/2004

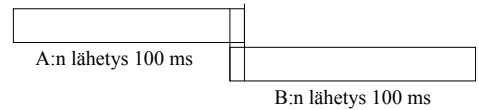
42

## Viipaloitu ALOHA (Slotted Aloha)

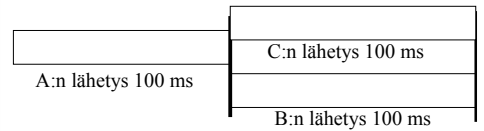
- lähetyisaika jaettu aikaviipaleiksi
- lähetyks voi alkaa vain aikaviipaleen alussa
- törmäykset täydellisiä
  - lähetykset samassa aikaviipaleessa
  - törmäysvaara-aika = yhden aikaviipaleen mittainen
- suorituskyky kaksinkertaistuu
  - maksimi ~ 37%
  - siis 37% tyhjiä, 37% onnistuneita, 26% törmäyksiä

2/18/2004

43



Törmäyksessä molemmat lähetykset menevät piloille ja lähetyisaikaa hukkaan 100 ms + 100 ms!



Törmäykset täydellisiä: Vaikka molemmat lähetykset menevät pilalle, niin lähetyisaikaa kuluu hukkaan vain 100 ms.

2/18/2004

44

## CSMA (Carrier Sense Multiple Access)

- toiminta
  - kuuntele linjaa ennen lähettämistä
  - jos linja vapaa lähetä (yleensä)
  - jos linja varattu odota satunnainen aika ja yritä uudelleen
- Suorituskyky:
  - törmäysvaara vain jos asemat lähettävät niin samanaikaisesti, että eivät siirtoviipeen vuoksi havaitse toista lähetyksestä
  - ongelma, jos siirtoviive on pitkä

2/18/2004

45

## CSMA/CD (Collision Detection)

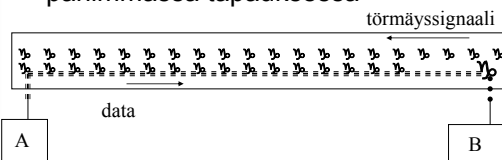
- keskeyttää lähettämisen heti, kun havaitsee törmäyksen tapahtuneen
  - törmäyksen aiheuttama hukka-aika pienenee
- 'epävarmuuden aika' on  $2\tau$ ,  $\tau$  on maksimi etenemisviive kahden aseman välillä
- jos törmäys
  - => havaitaan ja lopetetaan lähetyks
  - => yritetään uudestaan satunnaisen ajan kuluttua

2/18/2004

46

## Väylää kuunneltava

- pahimmassa tapauksessa



- => kehyksen lähetyksen minimikesto =  $2 \cdot$  etenemisviive väylällä

2/18/2004

47

## Vuorottavat protokollat

- Varausprotokollat

lähetyksvuorot varataan etukäteen

- varausvaihe
  - usein kilpaillaan varauksista
    - »törmäyksiä, mutta vähän
- lähetyksvaihe
  - kaikki varanneet lähettävät sanomansa
- hyvin paljon erilaisia versioita
  - etenkin satelliittiyhteyksille

2/18/2004

48



## ■ Pollaus (vuorokysely)

- isäntäasema antaa vuorotellen muille asemille lähetysoikeuden

## ■ Vuoromerkki

- asemilla kiertää vuoromerkki (token)
- asema saa lähettää vain kun sillä on vuoromerkki
- kun asema on lähettänyt tai sillä ei enää ole lähetettävää, se siirtää vuoromerkkin seuraavalle

2/18/2004

49

## 5.4. LAN-osoitteet ja ARP

### ■ (lähi)verkko-osoite

- fyysinen osoite
- MAC-osoite

### ■ Eetteriverkossa (sovitinkortissa)

- 48 bittiä
- joka kortissa oma ainutkertainen pysyvä numero (ROM-muistissa)

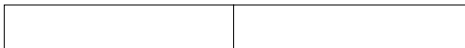
### ■ lähiverkkoon liitetyt laitteet ymmärtävät vain LAN-osoitteita

2/18/2004

50

2\*\*24 kpl

2\*\*24 kpl



Kaikkiaan 48 bittiä, joista 24 bittiä kertoo sovitinkortin valmistajan ja 24 bittiä identifioi sovitinkortin.

Kuudesta tavusta koostuva osoite: 1A-23-F9-CD-06-9B

Osoite on kiinteä ja säilyy, vaikka laite siirretään toiseen verkkoon.

Sovitin välittää omalle koneelleen vain omalle koneosoitteelleen osoitetut sanomat.

**LAN-yleislähetysosoite:** FF-FF-FF-FF-FF-FF

2/18/2004

51

## LAN-osoitteen käyttö

### ■ Jotta datagrammi voidaan lähettää lähiverkossa oikeaan koneeseen, on tunnettava koneen LAN-osoite

- Datagrammissa on vain IP-osoite!
  - IP-osoite => LAN-osoite
- Samankaltainen ongelma kuin selvitettyä domain-osoitetta vastaava IP-osoite.
  - DNS ratkaisee ongelman domain-osoite => IP-osoite

2/18/2004

52

## IP-osoite => LAN-osoitteeksi

### ■ ARP-taulu

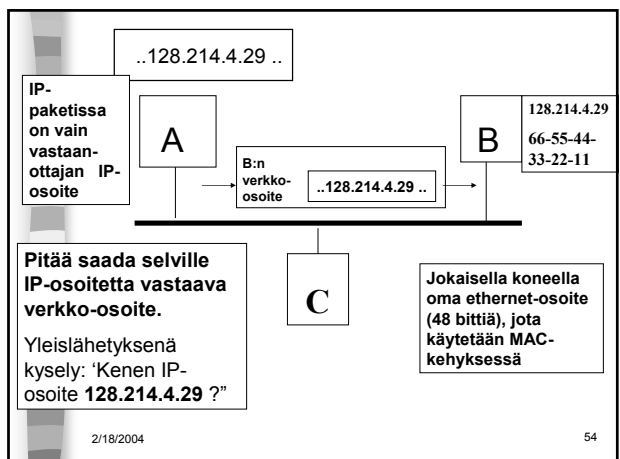
- IP-osoitteiden muuttamiseksi LAN-osoitteiksi
  - IP-osoite, sitä vastaava LAN-osoite, aikaleima
  - vanhentuneet tiedot katoavat taulusta (n. 20 minuuttia)

### ■ Entä, jos IP-osoitetta ei ole taulussa?

- Sovelluserroksella DNS, jolta kysyttiin.
- LAN:ssa kaikki asemat yleensä kuulevat kaikki lähetykset (yleislähetys).
- Hyödynnetään tätä ominaisuutta!

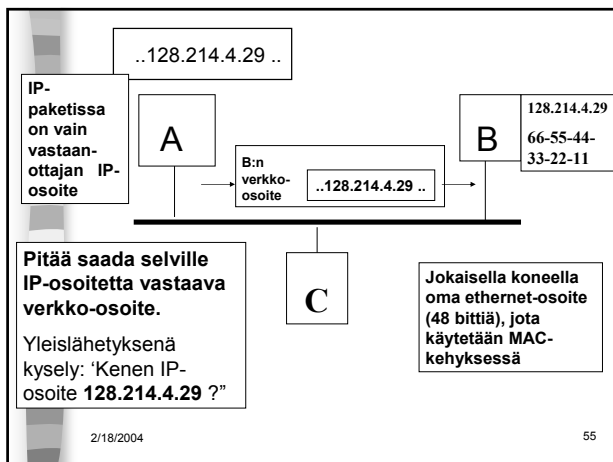
2/18/2004

53



2/18/2004

54



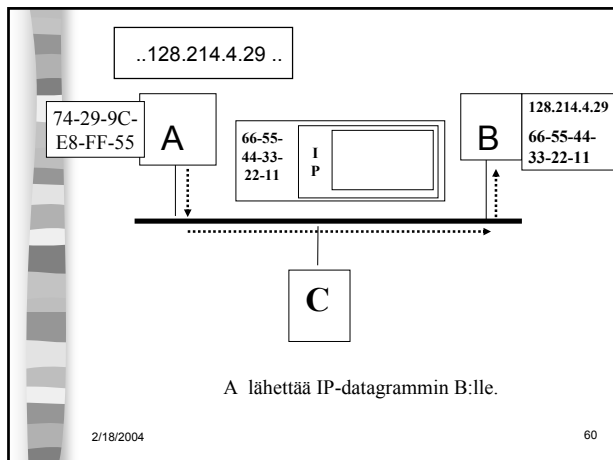
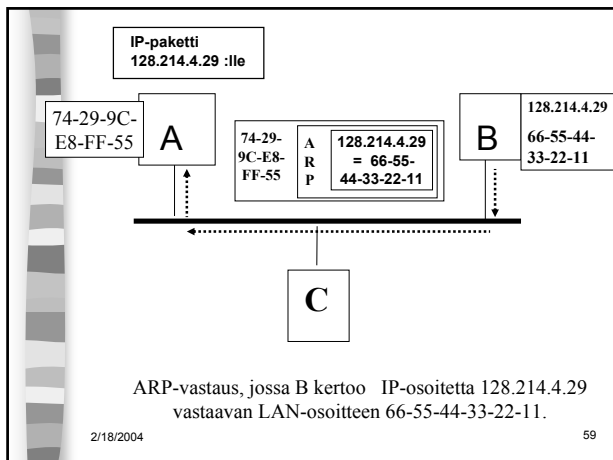
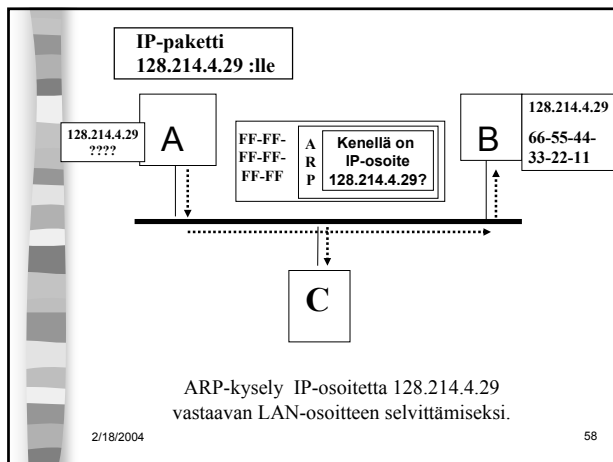
## ARP-protokolla (Address Resolution Protocol)

- IP-kerroksen protokolla, jolla selvitetään IP-osoitetta vastaava linkkerikerroksen osoite
  - esim. eetteriverkon 48-bittisiä osoitteita
- yleislähetys lähiverkkoon
  - "Kenellä on IP-osoite vv.xx.yy.zz?"
  - vastauksena osoitteen omistavan laitteen lähiverkko-osoite
    - ARP-paketteja: kysely ja vastaus

2/18/2004 56

- Jos A:lla ei ole tietoa ARP-taulussaan, niin A lähettää ARP-kyselyn yleislähetystenä
  - "Kenen IP-osoite on 128.214.4.29?"
  - Kone B, joka tunnistaa oman IP-osoitteensa, lähettää A:lle vastauksena ARP-paketin
    - "Koneen 66-55-44-33-22-11 IP-osoite on 128.214.4.29!"
  - A lähettää IP-paketin B:n LAN-osoitteella MAC-kehyyksessä.

2/18/2004 57

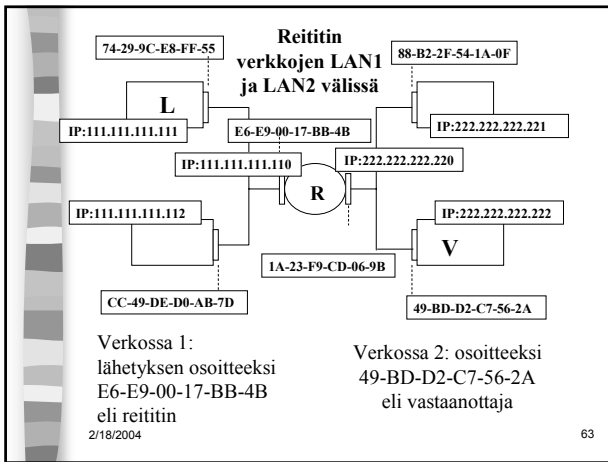


■ **optimointia:**

- kyselyn tulos välimuistiin
  - talletetaan muutaman minuutin ajan
    - » tyypillisesti 20 minuuttia
- kyselijä liittää omat osoitteensa kyselyyn
- alustettaessa jokainen laite ilmoittaa osoitteensa muille
  - kysyy omaa osoitettaan
  - jos tulee vastaus, niin konfigurointivirhe

**Lähtettäminen toiseen IP-verkkoon**

- Lähetetään ensin reitittimelle, joka reititystaulustaan tietää, minne porttiin eli mihin sovittimeen datagrammi on ohjattava
  - lähettävässä verkossa LAN-osoitteeksi reitittimen osoite
  - vastaanottavassa verkossa vastaanottavan koneen osoite
- LAN-osoitteet saadaan ARP-protokollan avulla



**5.5 Ethernet-lähiverkko**

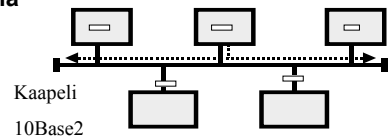
- Yleisin lähiverkkoteknologia
- IEEE:n standardoima LAN-verkko
  - Klassinen Ethernet: CSMA/CD (kuulosteluväylä)
  - Fast Ethernet, Gigabit Ethernet: yleensä kytkentäisiä kaksipisteyhteyksiä
  - 10 Gigabit Ethernet: kaksisuuntaisia, kaksipisteyhteyksiä

■ **Muita lähiverkkostandardeja**

- esim.
    - Token ring (vuororengas)
    - FDDI
    - WLAN (langaton lähiverkko)
- ei käsitellä tällä kurssilla**

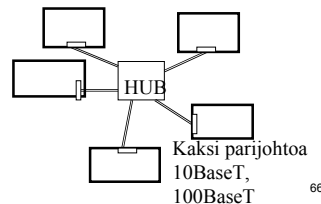
**Eetteriverkon rakenne**

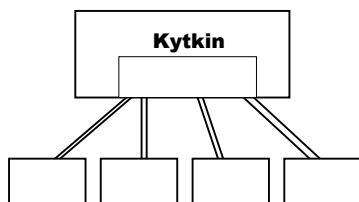
■ **väylä**



◆ **tähti**

- hub toimii toistimen tavoin





2/18/2004

67

## Kaapelit

### 10Base2 ohut kaapeli

- 10 => 10 Mbps
- Base => kantataajuus
- 2 => 200 m
- 10Base-T kierretty pari & central hub
  - helppo hallita, kallis, suosio kasvaa
- 10Base-F valokaapeli
  - kallis, luotettava, tehokas
- 100Base-T, 100 Base-F
  - Fast Ethernet
- 1000Base-T, 1000Base-X
  - Gigabit Ethernet

2/18/2004

68

## Lyhyet etäisyydet, pieni määrä laitteita

- sovittimesta keskittimeen (hub)
  - maks. 100 m
- segmentti
  - pituus maks. < 200 metriä,
    - syynä vaimeneminen
  - solmuja maks. 30 kpl
    - syynä CSMA/CD => liikaa törmäyksiä
  - maks. 5 segmenttiä voidaan yhdistää toistimilla
    - => ~1000 m, 150 laitetta
- valokuitua käytettäessä hieman pitemmät etäisyydet

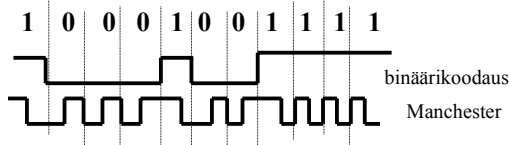
2/18/2004

69

## Signaalin koodaus (fyysinen kerros)

### ■ Manchester-koodaus

- tahdistus
  - ainajännitteen muutos keskellä bittiä
    - 1-bitti ylhäältä alas
    - 0-bitti alhaalta ylös
  - ei kellopulsseja
  - mutta lisää kaistanleveyttä



2/18/2004

70

## CSMA/CD

- jos väylä vapaa, lähetetään heti
  - muuten jäädään odottamaan ja lähetetään heti linjan vapauduttua
- aina kun on lähetetty, jäädään kuuntelemaan, onnistuiko lähetyks
- entä kun tapahtuu **törmäys** eli usea samanaikainen lähetyks
  - » jännite on suurempi kuin normaalisti
  - keskeytetään lähettäminen ja odotetaan satunnainen aika

2/18/2004

71

## Törmäyksen jälkeinen uudelleenlähetyks

### ■ Binary exponential backoff

- törmäyksen jälkeen aika jaetaan lokeroiksi
  - 51.2  $\mu$ s vastaten 512 bittiä eli 64 tavua
- 1. törmäyksen jälkeen asema odottaa satunnaisesti joko 0 tai 1 lokeron ajan ennen kuin yrittää uudelleen
- 2. törmäyksen jälkeen odotus on 0, 1, 2 tai 3 lokeroa
- n. törmäyksen jälkeen valitaan odotusaika väliiltä:
  - 0 –  $(2^{**n}) - 1$  lokeroa
    - 10. törmäyksen jälkeen väliä [0-1023] ei enää kasvateta
    - 16. törmäyksen jälkeen luovutaan ja ilmoitetaan 'asiakkaalle' (eli verkkokerrokselle) epäonnistumisesta

2/18/2004

72

- binäärinen eksponentiaalinen perääntymien on joustava
  - kuorma kasvaa eli törmäykset lisääntyvät => väli kasvaa
- vaihtoehtona kiinteä valintaväli
  - aina [0- 1023]
  - aina [0-1]
  - aina [a-n]
- kiinteän välin suorituskyky?
  - Pitkä väli turhaa odotusta
  - Lyhyt väli turhia törmäyksiä

2/18/2004

73

## MAC-protokolla

preamble	Destin. address	Source address	type	data	CRC
8 B	6 B	6 B	2 B	46-1500 B	4 B

## Ethernet-kehys

2/18/2004

74

## MAC-protokolla:

- tahdistuskuvio (preamble)
  - 7 tavua 1010101010 tahdistusta varten
  - kehyksen alku 10101011
- kohde- ja lähdeosoitteet
  - osoitteessa 6 tavua (tai 2 tavua)
  - 0xxxxx... yksilöosoite
  - 1xxxxx ... ryhmäosoite
  - 11111 .... kaikkia
  - yksi bitti: paikallinen vai globaali osoite

2/18/2004

75

## ■ Type

- kertoo käytetyn verkkoprotokollan tyyppin eli mille protokollalle kehyksen data luovutetaan
  - IP, ARP,
  - joku muu verkkoprotokola: AppleTalk, Novell IPX, ..

## ■ CRC

- 4 tavua

2/18/2004

76

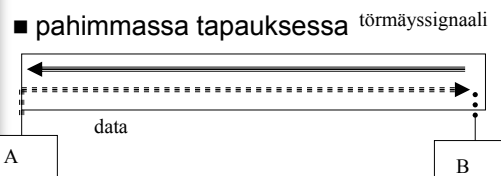
## kehyksen pituus

- 64-1500 tavua
  - kehyksen pituus vähintään 64 tavua
    - tarvittaessa täytettä (PAD)
- jotta lähettäjä ehtii havaita kehyksen törmäyksen
  - kehyksen lähetys ei saa päättyä ennen kuin alku on perillä ja mahdollinen törmäysääni kuuluu
    - alku perillä => loppukin onnistuu

2/18/2004

77

## Väylää kuunneltava



- => kehyksen lähetysten minimikesto:  $2 \times$  etenemisviive väylällä

2/18/2004

78

- 10 Mbps
  - LAN-pituus korkeintaan 2500 m
  - toistimia korkeintaan 4
  - lähetyksen kestettävä ainakin 51.2  $\mu$ s
  - eli 64 tavua

2/18/2004

79

## Ethernetin hyvät puolet

- yleisesti käytetty, yhteensopivuus aikaisempien Ethernet-versioiden kanssa
  - yksinkertainen protokolla, kevyellä kuormalla lähety sviive nolla
  - asemien lisääminen helppoa, hallinta yksinkertaista
  - passiivinen kaapeli, ei modeemia,
  - kukkaron ja tarpeen mukainen toteutus
    - Halpa perusmalli  $\leftrightarrow$  huippunopea
    - Hyvin erilaisia teknologioita
- => markkinajohtaja

2/18/2004

80

## Klassisen Ethernetin huonot puolet

- analoginen törmäyksen havaitseminen
- pienin kehys 64 tavua
  - => yleisrasitetta, jos sanomat lyhyitä
- epädeterministinen, ei prioriteetteja
- raskas kuorma
  - => törmäyksiä => suoritusteho laskee
- Nopeissa kytketyissä Ethernet-verkoissa ei ole törmäyksiä eikä epädeterministisyyttä

2/18/2004

81

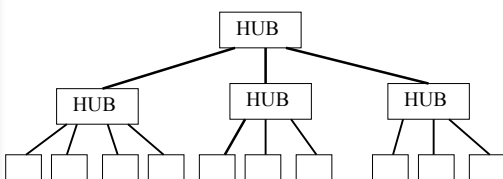
## 5.6 Keskitin (hub), silta (bridge) ja kytkin (switch)

- LAN-verkkojen yhdistäminen
  - keskittimillä (hub)
    - toistin, toimii perustasolla, käsittelee bittejä
    - lähettää vastaanottamansa bitit kaikille muille
    - yhteinen **törmäysalue** => vain pieniin verkkoihin
    - vain samanlaisiin verkkoihin
  - Silloilla (bridge) ja kytkimillä (switch)
    - linkkitason olioita
    - voivat periaatteessa yhdistää myös erilaisia verkkoja
      - mitä erilaisempia sen hankalampaa

2/18/2004

82

## Yhdistäminen keskittimillä



Yhteinen törmäysalue: vain yksi koneista voi samaan aikaan lähettää. Jos usea lähettää, tuloksena törmäys.

2/18/2004

83

## Keskitynyhdistämisen

- Etuja
  - voidaan yhdistää eri osastojen lähiverkot
  - suuremmat etäisyydet
  - rajoitetummat vikatilanteet
- Haittoja
  - sama kapasiteetti jaetaan useammalle
  - teknologialtaan erilaisia verkkoja ei voida yhdistää
  - vain rajallinen määrä laitteita

2/18/2004

84

## SILTA (Tuntumaton silta) (transparent bridge, spanning tree bridge)

### ■ tavoitteena tuntumattomuus

- 'plug and play'
  - ei mitään muutoksia laitteistoon, ohjelmistoon
  - ei reititystaulujen ja parametrien asettelua
  - ei vaikuta itse LANien toimintaan

### ■ tuntumaton silta

- vastaanottaa kaikki siihen kytketyiltä LANeilta tulevat kehykset
- joko hylkää tai ohjaa edelleen

2/18/2004

85

### ■ Tuntumaton silta

- tekee itse kaikki ohjausratkaisut
- silta alustaa itse itsensä
- silta sopeutuu dynaamisesti verkon muutoksiin

### ■ eri LANeista voi tulla sanomia yhtäaikaan

- talletetaan puskureihin

### ■ edelleen lähetettävistä sanomista valmistetaan niiden kohdeverkkoa vastaava kehys

2/18/2004

86

## Sillan portit

### ■ Lähiverkko liitetään siltaan **siltaportin** kautta

- yksinkertaisissa silloissa vain kaksi porttia
- monipuolisissa useita => kytkimiä (switch)

### ■ Portti

- MAC-piiri
  - noudattaa lähiverkon protokollaa
  - esim. CSMA/CD
- ohjelmisto
  - huolehtii alustuksesta
  - puskurin hallinnasta

2/18/2004

87

## Silta ohjaa kehykset toisiin LANeihin

### ■ siltojen siltataulut

	Laite-osoite	Portti	Laite-osoite	Portti
Jokaisella koneella oma yksikäsitteinen osoite = LAN-osoite	A	1	B	1
	B	1	C	1
	C	2	D	2
	D	2	H	3
	F	2		

Silta B1

Silta B2

2/18/2004

88

## Siltataulut

- Alkutilanteessa kaikkien siltojen siltataulut ovat tyhjiä.
- Siltataulua päivitetään aina, kun kehys saapuu.
- Vanhentuneet tiedot poistetaan.
  - ajastin laukeaa

2/18/2004

89

## Silta käsittelee kaikki kehykset:

**Kehys:** lähdeLAN X; kohdeLAN Y; tuloportti p;

### ■ Lähde ja kohde siltataulussa

- X ja Y samassa **portissa** => hylkää kehys
- X ja Y eri **porteissa** => lähetä eteenpäin
- päivitä X, p

### ■ Lähde ei taulussa

- lisää X, p, aika => silta oppii (backward learning)

### ■ Kohde ei taulussa

- lähetä Y kaikista muista porteista => tulvitus
- päivitä X, p

2/18/2004

90

## Tulvitus (flooding)

- tulvitus on ongelma
  - sanomat jäävät kiertämään silmukoissa
  - koko verkko tukkeutuu
- **siis silmukoita ei saa muodostua!**
  - eli verkon loogisen rakenteen pitää olla puu
  - muodostetaan verkolle ns. **virittävä puu** (spanning tree)

2/18/2004

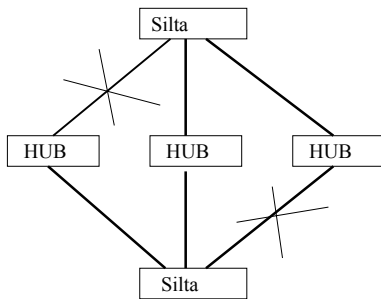
91

## Virittävä puu

- sillat muodostavat ja ylläpitävät
  - valitse juuri
    - silta, jolla pienin sarjanumero
  - valitse kustakin sillasta/ LAN:ista lyhin reitti juureen
    - => **virittävä puu**
      - muut sillat jäävät käyttämättä
  - tulvitus vain **virittävän puun siltoja pitkin**

2/18/2004

92



2/18/2004

93

## Siltojen edut

- verkkojen ja asemien määrää helppo kasvattaa
- erilaisia lähiverkkoa
- sillat eivät näy ylemmille kerroksille
- voidaan kerätä tietoja ja säädellä pääsyä
- luotettavuus ja suorituskyky kasvaa

2/18/2004

94

## Siltojen haitat

- sillat puskuroivat ja aiheuttavat viivettä
- ei vuonsäätelyä => sillan kapasiteetti voi ylittyä
- kehysrakenteen muuttaminen => virheitä jää havaitsematta
- **Yleisesti edut selvästi suuremmat kuin haitat**

2/18/2004

95

## Kytkein (switch)

- Erittäin suorituskykyisiä, moniporttisia siltoja
  - silloissa muutamia portteja
  - kytkimissä kymmeniä portteja (liitäntöjä)
  - portit voivat olla erinopeuksisia
  - kaksisuuntainen lähetys (full-duplex)
  - verkonhallintapiirteitä, **suorakytkentä** (cut-through)
- Koneet voidaan liittää suoraan kytkimeen
  - kukin kone voi lähettää täydellä nopeudella
  - ei törmäyksiä!

2/18/2004

96



## Erittäin nopeat lähiverkot

(High-speed LANs)

- nopeus >> 10 Mbps, 100 Mbps - 10 Gbps
- eri ratkaisuja
  - **Fast Ethernet, Gigabit Ethernet, 10 Gigabit Ethernet**
  - FDDI, HIPPI, WLAN, atm, jne
  - Näitä ei käsitellä kurssilla tarkemmin!

2/18/2004

97

## 5.7. PPP-protokolla

- Linkkitason protokollia on useita
  - **HDLC** (High-level Data Link Control)
    - useita, enemmän tai vähemmän toisistaan poikkeavia yhteensopimattomia versioita
    - ei käsitellä kurssilla
  - **PPP** (Point-to-Point Protocol)
    - soittoyhteys modeemiin tai ISDN:n kautta tietokoneeseen
    - yleisimmin käytettyjä linkkiprotokollia

2/18/2004

98

## LLC (Logical Link Control)

- Erilaisia LAN-verkkoja
- **vuonvalvonta, virhevalvonta, yhtenäinen rajapinta erilaisiin verkkoihin**
  - ~ OSI-malli, HDLC
- **Palvelut:**
  - epäluotettava datasäikepalvelu,
  - kuittava datasäikepalvelu,
  - luotettava yhteydellinen palvelu

verkkokerros
LLC
MAC
peruskerros

2/18/2004

99

## PPP (Point-to-Point Protocol)

- IETF:n (Internet Engineering Task Force) vaatimuksia
  - hyvin toimiva kehystys
  - kehysten virhetarkistus (virheellinen kehys tuhotaan!)
  - havaitsee, jos yhteys ei toimi ja ilmoittaa tästä verkkokerrokselle
  - useat verkkokerroksen protokollat voivat käyttää
  - verkko-osoitteista sopiminen: mm. IP-osoitteet neuvoteltavissa yhteyden muodostuksen aikana
  - autentisointi mahdollista
  - ei vuonvalvontaa

2/18/2004

100

## PPP-kehys

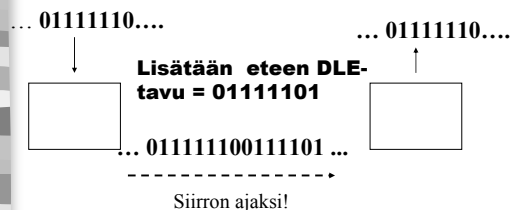
Tavuja 1 1 1 1-2 vaihtelee 2-4 1

01111110	osoite	kontrolli	protokolla	data	CRC	01111110
----------	--------	-----------	------------	------	-----	----------

- **lipputavu 01111110,**
  - tavunlisäys (byte stuffing) DLE = 01111101
- **osoitekenttä aina 11111111 (=yleislähetys)**
- **kontrollikenttä aina 00000011**
  - osoite- ja kontrollikenttä voidaan jättää kokonaan pois
- **protokolla: mille protokollalle data on tarkoitettu**
  - esim. IP, IP:n Control Protocol, PPP:n Link Control Protocol
- **data: sisältää ylempälle protokollalle tarkoitettua dataa**
  - maksimi sovitaan, oletusmaksimi 1500 tavua
- **CRC: tarkistusbitit;**

## Tavunlisäys

Jos datassa on lipputavu 01111110 ?



Entä, jos datassa on ..0111101 ...?

2/18/2004

102

## ■ LCP (Link Control Protocol)

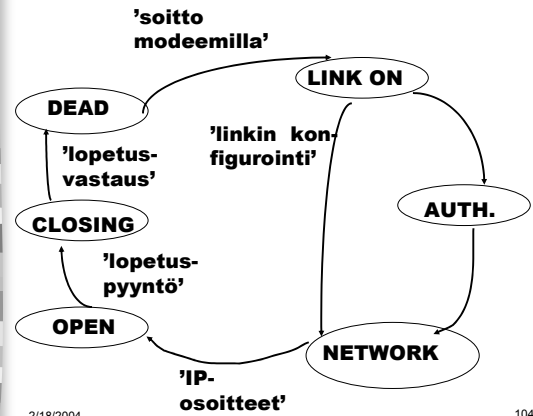
- muodostaa ja testaa linjayhteyksiä
- neuvottelee yhdeyden ominaisuuksista
- purkaa yhteyden, kun sitä ei enää tarvita
- vrt. TCP-yhteys

## ■ NCP (Network Control Protocol)

- neuvottelee verkkokerroksen optioista
- oma NCP kullekin verkkoprotokollalle
- TCP/IP: tärkein tehtävä IP-osoitteen antaminen päätteelle dynaamisesti

2/18/2004

103



2/18/2004

104

## Yhteydenotto PPP:llä

- soitto modeemilla reitittimeen
  - fyysinen yhteys
- PPP-parametrien valinta
  - LCP-paketteja vaihtamalla
- verkkokerroksen konfigurointi
  - TCP/IP: IP-osoitteen antaminen PC:lle
  - PC => tilapäinen Internet isäntäkone
- PC voi lähettää ja vastaanottaa tavallisen isäntäkoneen tapaan

2/18/2004

105

## Yhteyden purku

- NCP purkaa verkkoyhteyden ja vapauttaa IP-osoitteen
- LCP purkaa siirtoyhteyserroksen

2/18/2004

106

## Linjayhteyden muodostus

- Dead
  - ei kantoaaltoa, ei peruserroksen yhteyttä
- Link (Established)
  - peruserroksen yhteys muodostettu
  - sovitaan LPC-optioista
- Authenticate
  - osapuolet varmistuvat toistensa identiteetistä
- Network
  - NCP konfiguroi verkkokerroksen

2/18/2004

107

- Open
  - tiedonsiirto voi alkaa
- Closing
  - kun tiedonsiirto suoritettu => lopetustilaan
  - tästä palataan alkutilaan lopettamalla kantoaalto

2/18/2004

108

# LPC-pakettityypit

## ■ optioista ja niiden arvoista sopiminen

### – Configure-

- request ehdotettuja optioita ja arvoja
- ack kaikki hyväksytään
- nak optioita, joita ei voida hyväksyä
- reject optioita, joista ei voida neuvotella

2/18/2004

109

## ■ linjan sulkeminen

### – Terminate-

- request linjan sulkemispyyntö
- ack OK, linja suljetaan

2/18/2004

110

## ■ tuntemattomat sanomat

- Code-reject tuntematon pyyntö
- Protocol-reject tuntematon protokolla

## ■ linjan testaus

- Echo-request palauta tämä kehys
- Echo-reply tässä kehys takaisin
- Discard-request hylkää tämä testisanoma

2/18/2004

111

## Yhteenveto:

- Sovelluskerros: sovelluksen tarpeet
  - HTTP, DNS, SMTP
- Kuljetuskerros: sanomien kuljetus prosessin välillä luotettavasti
  - TCP: virheet, vuon- ja ruuhkanvalvonta;
  - UDP
- Verkkokerros: reititys koneiden välillä
  - IP, osoitteet, reititysprotokollat, reititin
- Siirtoyhteyskerros: kahden solmun välillä
  - MAC: CSMA/CD, CDMA; PPP
  - Ethernet, silta

2/18/2004

112

