

5. Siirtoyhteyskerros

linkkikerros (Data Link Layer)

- yhtenäinen linkki solmusta solmuun

- bitit sisään => bitit ulos

- ongelmia:

- siirtovirheet
 - havaitseminen
 - korjaaminen
- solmun kapasiteetti
 - vuonvalvonta
- yhteisen kanavan käyttö



9/1/2003

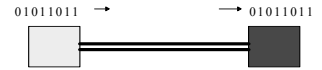
1

5.1. Kaksipisteyhteydet

Virhevalvonta

- * virheiden havaitseminen
- * virheiden korjaus

Vuonvalvonta



9/1/2003

2

Bittivirta <=>kehysiä

- tavoite

- bittivirheiden hallinta
 - muuttuu
 - katoaa
 - monistuu

- bittivirta kehyksinä

- kehys tarkistettavissa

- tarkistustietoa

9/1/2003

3

Kehysten kuljetus

- tavoite

- kaikki kehykset
- kukin kehys virheettömästi
- lähetyssjärjetyksessä
- 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

9/1/2003

4

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

9/1/2003

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

9/1/2003

6

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 uudelleenlähety

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
 - satelliittiyhteyksissä, avaruusluotaimissa
- Esimerkkejä
 - Hamming-pariteettitarkistus (Tito-kurssilla)
 - pystyy korjaamaan yhden virheellisen bitin
 - virheryöpy, jos se jaetaan yhden bitin virheeksi
 - Reed-Solomon -koodit
 - lohkokodeja, jotka pystyvät korjaamaan virheryöppyjä

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 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

Horizontaaliset ja vertikaaliset pariteetit

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

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

9/1/2003

13

Virheiden havaitseminen

- Ei löydy lyhyitä virheryöppyjä, joissa neljä bittiä vaihtuu sopivasti

```

1 0 0 1 0 1 0
0 1 0 1 0 1 0
1 1 1 0 0 0 1
1 0 0 0 1 1 1
    
```

9/1/2003

14

Internetin tarkistussumma

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

9/1/2003

15

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

data	tarkistusbitit
1 0 0 1 1 0 1 1 0	□ □ □ □ □ □ □ □
K bittia	L bittia

9/1/2003

16

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
    
```

Modulo 2-
aritmetiikka:
 $1+1 = 0$ (XOR)

Lähetetään: 101110 011

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

tarkistusbitit

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 $< \text{tai} =$ kuin virittäjän
- useimmat virheryöpyt, joiden pituus on suurempi
 - CRC-32: $P\{\text{ryöppy} > 33 \text{ havaitaan}\} = 0.9999999999$

– Huom

- » Arvioinneissa lähtökohdana ollut täysin satunnainen bittien jakautuminen, mutta todellisuudessa näin ei ole!
- » Joten havaitsemattomien virheiden määrä on arvioitua suurempi.

9/1/2003

18

Vuonvalvonta

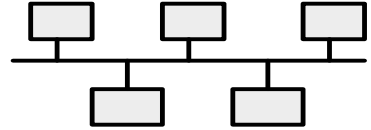
- **Liukuva ikkuna**
 - ikkunan koko rajoittaa lähettämistä
 - » jos kehyksen numero ei ole ikkunassa, sitä ei oteta vastaan
 - kuitaus siirtää ikkunaa eteenpäin
- **stop-sanoma**
 - Receive not ready

9/1/2003

19

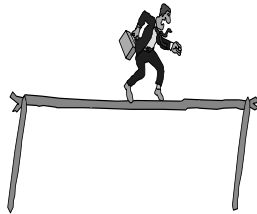
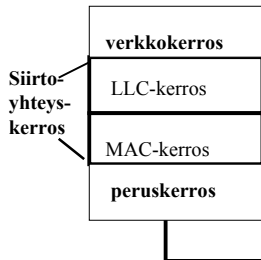
5.3. Yhteiskäyttöinen kanava

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



MAC = Medium Access Control

LLC = Logical Link Control



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'

9/1/2003

22

Törmäys

- yksi yhteinen kanava lähettäjiille
 - 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 törmäyksiä (=hukkaan) menevää aikaa
 - törmäykset täydellisiä

Lähetyskanavan 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. satelliitilähetykset

Kanavan kuuntelu

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

9/1/2003

26

Yleislähetysprotokollia

Esimerkkejä:

- **CSMA/CD** (Aloha, CSMA)
 - mm. Ethernet-verkossa käytetty kilpailuprotokolla
- **CDMA**
 - radiolinjoilla käytetty koodinjakoon perustuva protokolla

9/1/2003

27

ALOHA

- Hawaiilla, 70-luvulla radiotietä varten
- puhdas ALOHA:
 - asema lähettää aina, kun sillä on lähetettävää
 - ja samalla kuuntelee, onnistuiko lähetys
 - lähiverkossa törmäys havaitaan 'heti', sillä siirtoviive pieni
 - toisin satelliitilla!
 - jos törmäys, niin lähettäjä odottaa satunnaisen ajan ja yrittää uudelleen
 - maksimaalinen tehokkuus ~18%

Viipaloitu ALOHA

- lähetysaika jaettu aikaviipaleiksi
- lähetys 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

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 siirtoviiveen vuoksi havaitse toista lähetystä
 - ongelma, jos siirtoviive on pitkä

CSMA-protokollat

- Useita versioita, jotka hieman eroavat toisistaan
 - miten toimitaan, kun kanava varattu?
 - jäädytään odottamaan ja lähetetään heti kanavan vapauduttua => jos useita odottajia, tulee varmasti törmäys
 - luovutaan ja yritetään uudestaan satunnaisen ajan kuluttua => hukkaa lähetysvuoroja
- viipaloitu aika vai ei?
- vaikka kanava on vapaa, ei silti aina lähetetä
 - lähetys vapaalle väylälle todennäköisyydellä p!

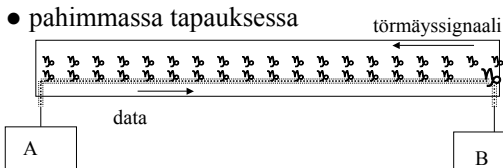
CSMA/CD (Collision Detection)

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

9/1/2003

32

Väylää kuunneltava

- pahimmassa tapauksessa
- 
- data
- törmäyssignaali
- A
- B
- => kehyksen lähetysminimikesto:
 $2 \cdot \text{etenemisviive väylällä}$

9/1/2003

33

Varausprotokollat

- ei törmäyksiä!
- lähetysvuorot varataan etukäteen
- varausvaihe
 - usein kilpaillaan varauksista
 - törmäyksiä, mutta vähän
- lähetysvaihe
 - kaikki varanneet lähettävät sanomansa
- hyvin paljon erilaisia versioita
 - etenkin satelliittiyhteyksille

Vuorotteluprotokollat

- Pollaus (vuorokysely)
 - isäntäasema antaa vuorotellen muille asemille lähetysluvan
- 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

9/1/2003

35

Kanavan jakoprotokollat

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

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)

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

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

9/1/2003

39

- 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
 - kerrottuna bitin aikasirujen lukumäärällä

9/1/2003

40

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

9/1/2003

41

5.4. LAN-osoitteet ja ARP

- (lähi)verkko-osoite
 - fyysinen osoite
 - MAC-osoite
- Eetteriverkossa (sovitinkortissa)
 - 48 bittiä
 - joka kortissa oma ainutkertainen pysyvä numero
- lähiverkkoon liitetyt laitteet ymmärtävät vain LAN-osoitteita

9/1/2003

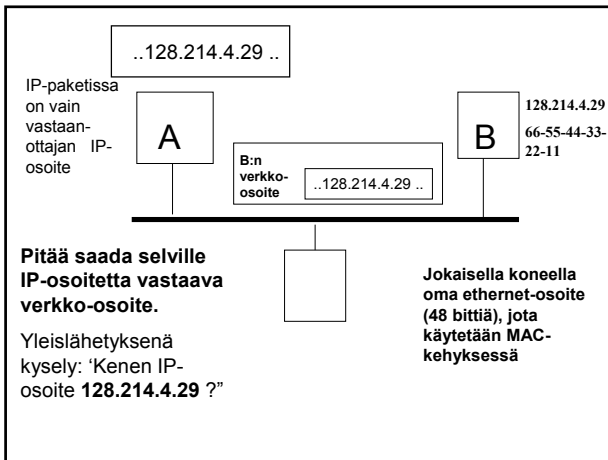
42

IP-osoite => LAN-osoitteeksi

- ARP-taulu
 - IP-osoitteiden muuttamiseksi LAN-osoitteiksi
 - » IP-osoite, sitä vastaava LAN-osoite, aikaleima
 - vanhentuneet tiedot katoavat taulusta
- Entä, jos IP-osoitetta ei ole taulussa?
 - Sovelluskerroksella DNS, jolta kysyttiin.
 - LAN:ssa kaikki asemat yleensä kuulevat kaikki lähetykset (yleislähetys).
 - Hyödynnetään tätä ominaisuutta!

ARP-protokolla (Address Resolution Protocol)

- IP-kerroksen protokolla, jolla selvitetään IP-osoitetta vastaava siirtoyhteyskerroksen 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



- Jos A:lla ei ole tietoa ARP-taulussaan, niin A lähettää ARP-kysely yleislähetyskyselynsä
 - » "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-kehyskessä.

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

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**

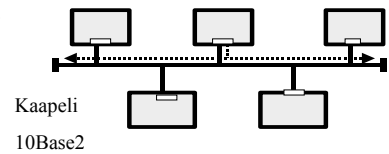
– esim.

- **Token ring (vuororengas)**
- **FDDI**
- **WLAN (langaton lähiverkko)**

ei käsitellä tällä kurssilla

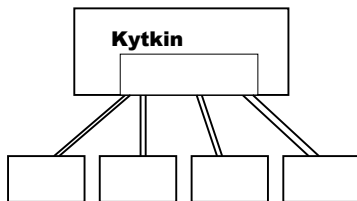
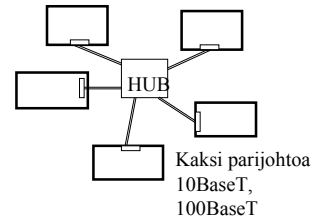
Eetteriverkon rakenne

- väylä



- ◆ **tähti**

- hub toimii toistimen tavoin



**Kytchentäinen,
kaksisuuntainen Ethernet:
Ei törmäyksiä**

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

Lyhyet etäisyydet, pieni määrä laitteita

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

Signaalin koodaus

- **Manchester-koodaus**

– tahdistus

» jännitteen muutos keskellä bittiä

- ei kellopulssia
- mutta lisää kaistanleveyttä

CSMA/CD

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

9/1/2003

55

Törmäyksen jälkeinen uudelleenlähetys

• Binary exponential backoff

- törmäyksen jälkeen aika jaetaan lokeroiksi
 - 51.2 μ 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 väliltä:
 $0 - 2^{*n} - 1$ lokeroa
 - 10. törmäyksen jälkeen väliä [0-1023] ei enää kasvateta
 - 16. törmäyksen jälkeen luovutetaan ja ilmoitetaan 'asiakkaalle' (eli verkkokerrokselle) epäonnistumisesta

- **binäärinen eksponentiaalinen perääntymien on joustava**
 - kuorma kasvaa => väli kasvaa
- **vaihtoehtona kiinteä valintaväli**
 - » aina [0- 1023]
 - » aina [0-1]
 - » aina [a-n]
 - entä suorituskyky?

9/1/2003

57

Ehernet-kehys

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

9/1/2003

58

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

9/1/2003

59

- **Type**
 - » kertoo käytetyn verkkoprotokollan tyyppin eli mille protokollalle kehyksen data luovutetaan
 - IP, ARP,
 - joku muu verkkoprotokola: AppleTalk, Novell IPX, ..
- **CRC**
 - » 4 tavua

9/1/2003

60

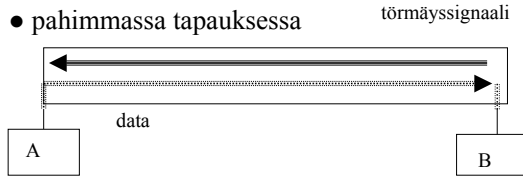
kehiksen pituus

- **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

9/1/2003

61

Väylää kuunneltava



- => kehyksen lähetysajan minimikesto:
 $2 * \text{etenemisviive väylällä}$

9/1/2003

62

- **10 Mbps**
 - LAN-pituus korkeintaan 2500 m
 - toistimia korkeintaan 4
 - lähetysajan kestettävä ainakin 51.2 μ s
 - eli 64 tavua

9/1/2003

63

Ethernetin hyvät puolet

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

9/1/2003

64

Klassisen Ethernetin huonot puolet

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

9/1/2003

65

5.6 Keskittin (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 ja kytkimillä
 - » linkkitason olioita
 - » voivat **periaatteessa** yhdistää myös erilaisia verkkoja
 - mitä erilaisempia sen hankalampaa

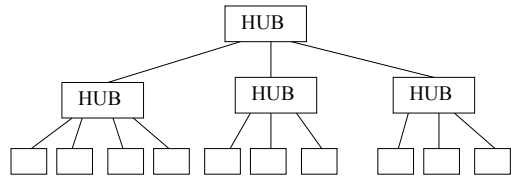
Käyttötarpeita

- osastoverkot
- maantiede: hajautus
- etäisyydet: yhdistäminen
- kuormituksen jakaminen
- häiriöiden rajoitus paikalliseksi
- suojaus: lähiverkkojen looginen eristäminen

9/1/2003

67

Yhdistäminen keskittimillä



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

9/1/2003

68

Keskityhdistämisen

- 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

9/1/2003

69

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

9/1/2003

70

- 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

9/1/2003

71

Sillan portit

- Lähiverkko liitetään siltaan **portin** kautta
 - yksinkertaisissa silloissa vain kaksi porttia
 - monipuolisissa useita => kytkimiä (switch)
- Portti
 - MAC-piiri
 - noudattaa lähiverkon protokollaa
 - esim. CSMA/CD
 - ohjelmisto
 - huolehtii alustuksesta
 - puskurin hallinnasta

Sillat ohjaavat kehykset toisiin LANeihin

• siltojen siltataulut

Jokaisella laitteella oma yksikäsitteinen osoite

laite-osoite	portti
A	1
B	1
C	2
D	2
F	2

Silta B1

Laitte-osoite	portti
B	1
C	1
D	2
H	3

Silta B2

Siltataulut

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

Silta käsittelee kaikki kehykset:

Kehys: lähdeLAN X; kohdeLAN Y; tuloportti I;

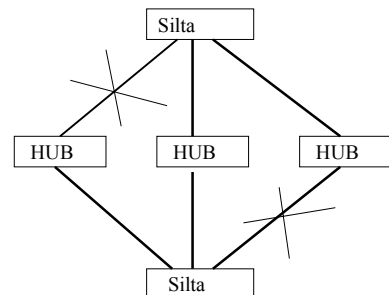
- Lähde ja kohde siltataulussa
 - X ja Y samassa **portissa** => hylkää kehys
 - X ja Y eri **porteissa** => lähetä eteenpäin
 - päivitä X, I
- Lähde ei taulussa
 - lisää X, I, aika => silta oppii (**backward learning**)
- Kohde ei taulussa
 - lähetä Y kaikista muista porteista => tulvitus
 - päivitä X, I

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)

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**



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

Siltojen haitat

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

Kytkin (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!

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!

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 modeemin tai ISDN:n kautta tietokoneeseen
 - yleisimmin käytettyjä linkkiprotokollia

LLC (Logical Link Control)

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

verkkokerros
LLC
MAC
peruserros

PPP (Point-to-Point Protocol)

- IETF:n 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

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 ?



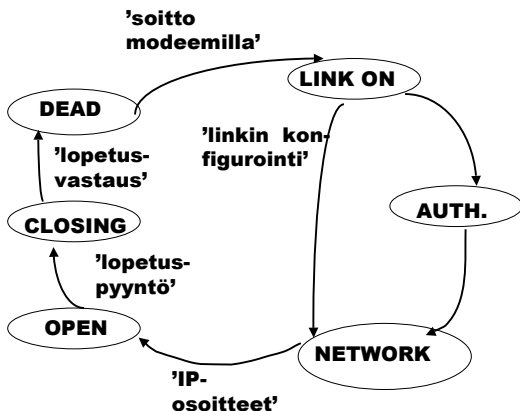
Entä, jos datassa on ..0111101 ...?

LCP (Link Control Protocol)

- » muodostaa ja testaa linjayhteyksiä
- » neuvottelee yhteyden 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



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

Yhteyden purku

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

Linjayhteyden muodostus

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

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

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
- linjan sulkeminen
 - Terminate-
 - » request linjan sulkemispyyntö
 - » ack OK, linja suljetaan

- 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

Yhteenveto

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

Kiitos kestävydestä!

