

Subnivelurile MAC și LLC

Nivelul legătură de date este împărțit în două subniveluri cu roluri diferite:

1. Subnivelul MAC (Media Access Control): - acest subnivel asigură accesul ordonat și controlat la mediu. Aceasta înseamnă spre exemplu că două stații nu pot transmite în același timp, iar erorile cauzate de încercările de a transmite simultan sunt detectate. Acest subnivel este dependent de tehnologie.
2. Subnivelul LLC (Logical Link Control): - acest subnivel are scopul de a asigura comunicarea între nivelul legătură de date și nivelul superior. Acest subnivel este absolut necesar pentru a ascunde tehnologiile LAN pe care se bazează rețeaua de nivelurile superioare. Acest subnivel este deci independent de tehnologie, oferind nivelurilor superioare funcții generice pentru transmisie, funcții ce sunt același pentru orice variații ale nivelului fizic și subnivelului MAC.

Niv. leg. de date	LLC
	MAC
Nivel fizic	

OSI – TCP/IP

Alocarea dinamică a canalului:

Nici una din metodele statice de alocare a canalului nu funcționează bine în condiții de trafic în rafală.

La baza întregii activități din acest domeniu stau câteva ipoteze-cheie.

1. **Modelul stațiilor:** Acest model constă din N stații independente, pe fiecare existând un proces care generează cadre de transmis (Probabilitatea de generare a unui cadru într-un interval de lungime Δt este $\lambda \Delta t$, unde $\lambda = \text{const}$, rata sosirilor de cadre noi). Odată ce a fost generat un cadru, stația se blochează și nu mai face nimic până la transmiterea cu succes a cadrului.
2. **Ipoteza canalului unic:** Există un singur canal accesibil pentru toate comunicațiile. Toate stațiile pot transmite prin el și recepționa de la el. În ceea ce privește partea hardware, toate stațiile sunt echivalente, deși protocolul software le poate acorda priorități diferite. (Aceasta este de fapt esența problemei, nu există mijloace externe de comunicare, stațiile nu pot cere permisiunea de a transmite)
3. **Ipoteza coliziunii:** Dacă două cadre sunt transmise simultan, ele se suprapun, iar semnalul rezultat va fi neglijabil. Acest eveniment se numește coliziune. Toate stațiile pot detecta coliziuni. Un cadru implicat într-o coliziune trebuie retransmis ulterior. Nu există alte erori în afara celor generate de coliziuni.
4. **Timp continuu:** Transmisia cadrelor poate surveni în orice moment. Nu există un ceas comun, care să împartă timpul în intervale discrete.

5. **Timp discret:** Timpul este împărțit în intervale discrete (cuante).
Transmisia cadrelor pornește mereu la începutul unei cuante. O cantă poate conține 0, 1, sau mai multe cadre, corespunzător unei cuante de așteptare, unei transmisiuni efectuate cu succes sau respectiv unei coliziuni.
6. **Detecția purtătoarei:** Stațiile pot afla dacă un canal este liber sau nu înainte de a încerca să-l utilizeze. Dacă el este deja ocupat, nici o stație nu va încerca să-l utilizeze până când nu se va elibera.
7. **Nedetecția purtătoarei:** Stațiile nu pot afla starea canalului înainte de a încerca să-l utilizeze. Ele pur și simplu încep să transmită. Abia după aceea vor putea determina dacă transmisia s-a efectuat cu succes sau nu.

Protocoloale cu acces multiplu:

ALOHA pur:

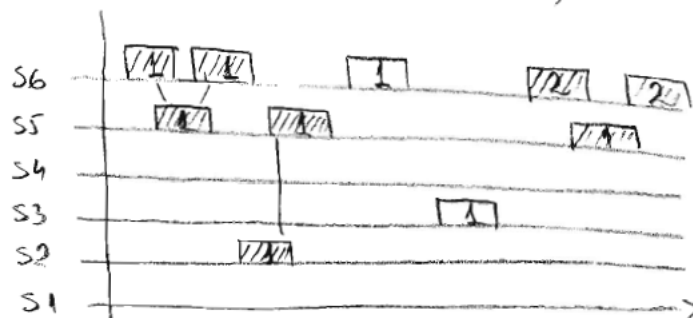
Într-un sistem ALOHA stațiile sunt lansate să transmită ori de câte ori au date de transmis. Întrucât este foarte probabil să apară coliziuni fiecare stație fie să-și monitorizeze propria transmisie fie să aștepte primirea unei confirmări. Prin compararea codului transmis cu cel recepționat sau prin prezența/absența confirmării, stația emițătoare poate determina dacă transmisia s-a efectuat sau nu cu succes. Dacă transmisia s-a soldat cu un eșec, cadrul este retransmis după un interval aleator pentru a reduce probabilitatea unei noi coliziuni.

Avantaje:

- eficiență mai bună decât în cazul alocării statice, atunci când există un număr mare de stații cu profil de trafic în rafală
- scalabilitate (adaptare la varierea numărului de stații)

Dezavantaje:

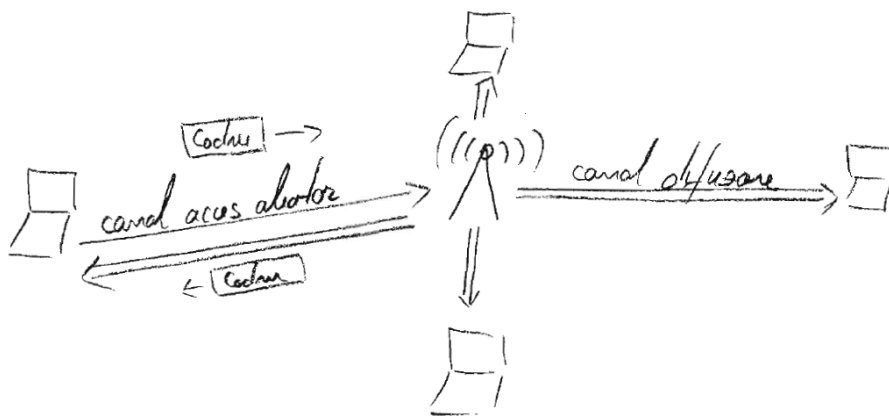
- eficiența maximă a utilizării canalului 18,4%
- sunt necesare buffer-e pentru retransmisie



Povestea:

- modem-uri radio 9600 bps, Hawaii 400km
- două canale:
 - canal de difuzare (100kHz)
 - canal de acces aleator (100kHz)

-topologie logică stea

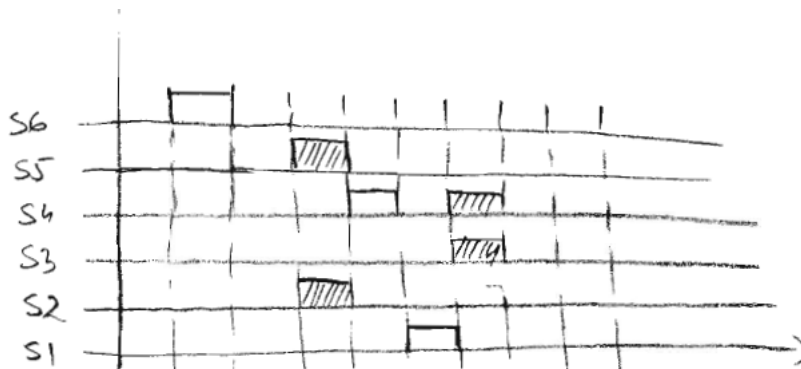


- existența a două canale reduce probabilitatea apariției unor coliziuni, întrucât pe canalul de difuzare unde există un singur emițător nu pot apărea coliziuni.

ALOHA cuantificat:

În acest caz timpul este împărțit în intervale discrete. Lungimea unui interval va fi egală cu timpul necesar transmiterii unui cadru (se consideră cadre de lungime fixă). O metodă de obținere a sincronizării ar fi ca o stație specială să emită un "bip" la începutul fiecărui interval. Unei stații nu îi mai este permis să transmită oricând (ca în situația anterioară), ea trebuind să aștepte începutul următoarei cuante de timp. În cazul în care apar coliziuni nu vor mai exista suprapuneri parțiale ci doar complete. Acest fapt duce la dublarea eficienței. În cazul în care apare o coliziune se așteaptă un număr aleator de sloturi după care se transmite din nou. Cea mai bună performanță la care ne putem aștepta este de:

- 37% cuante neutilizate
- 37% cadre transmise cu succes
- 26% coliziuni



Ethernet:

Protocolul de acces multiplu folosit CSMA/CD

Nodurile dintr-o rețea locală Ethernet sunt interconectate printr-un canal cu difuzare (broadcast), ceea ce înseamnă că un cadru transmis va fi recepționat de toate stațiile din rețea.

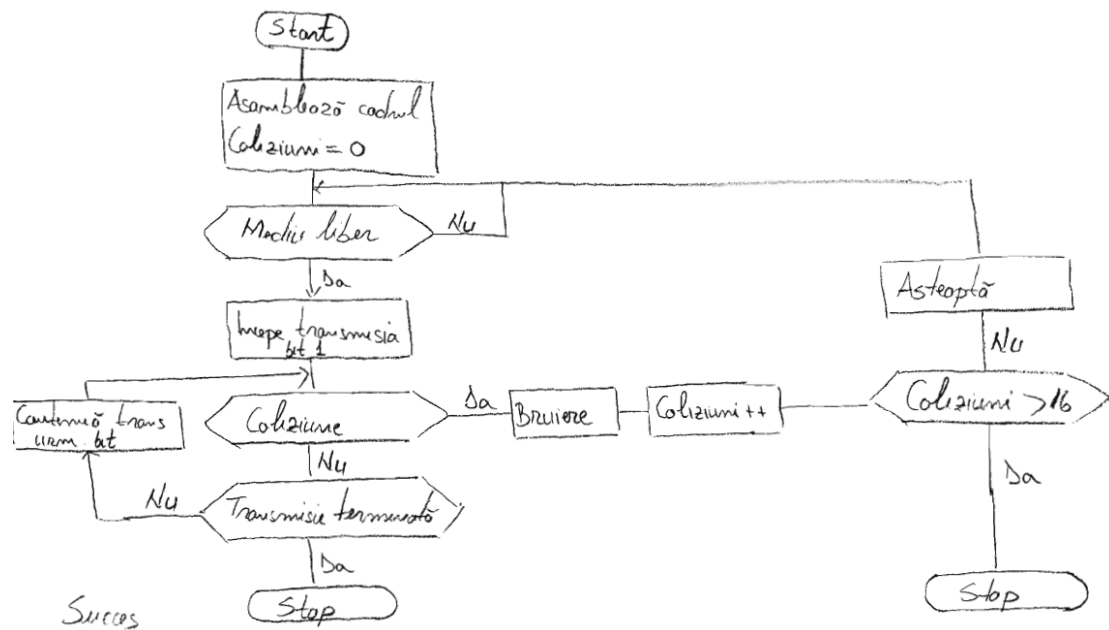
Algoritmul CSMA/CD (principiu):

1. O stație ar putea transmite oricând (nu se folosesc sloturi)
2. O stație nu va transmite niciodată dacă sesizează că altă stație transmite
3. O stație va abandona imediat propria transmisie imediat ce detectează că o altă stație transmite
4. Înainte de a încerca să transmită, o stație va aștepta o perioadă aleatoare (mică în raport cu timpul de cadre)

Algoritmul CSMA/CD în cazul Ethernet:

Fiecare interfață (adaptor) de rețea anulează protocolul CSMA/CD fără coordonare (explicită) cu celelalte adaptoare din rețea. Într-un adaptor specific protocolul se desfășoară în felul următor:

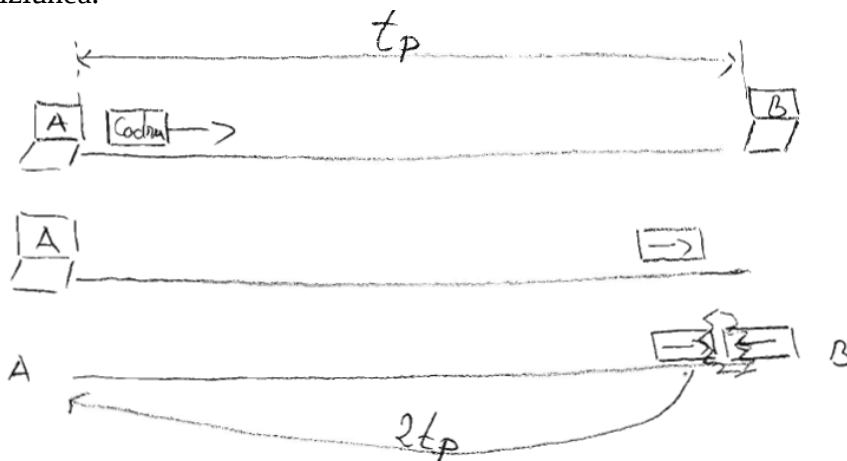
1. Adaptorul primește un pachet de la nivelul rețea, îl încapsulează și stochează cadrul în buffer-ul de transmisie.
2. Dacă detectează mediul ca fiind liber (evaluând nivelul de energie), începe să transmită cadrul. Dacă adaptorul detectează mediul ca fiind ocupat, așteaptă până la eliberarea acestuia (plus încă câteva sute de uS) după care începe să transmită cadrul.
3. Pe parcursul transmisiei, adaptorul monitorizează prezența uni eventual alt semnal provenit de la alte adaptoare. Dacă se reușește transmiterea întregului cadru fără detectarea unui alt semnal, atunci se consideră că acel cadru a fost transmis cu succes.
4. Dacă adaptorul detectează energie în timpul transmisiei, întrerupe transmisia și transmite în schimb un semnal de bruieră de 48 de biți.
5. După transmiterea semnalului de bruieră, adaptorul trece într-o fază de regresie exponențială binară. Mai exact, când se transmite un cadru, implicat în n coliziuni, adaptorul alege o valoare aleatoare a lui K din mulțimea $\{0, 1, 2, \dots, 2^m - 1\}$ unde $m = \min(n, 10)$. Adaptorul așteaptă apoi $512 \times K$ intervale de bit ($1\text{bit} = 0.1\mu\text{S}$) și revine la pasul 2.



În concluzie Ethernet-ul pune la dispoziție un serviciu neconfirmat fără conexiune.

Comentarii:

- Scopul semnalului de bruieră este ca toate stațiile care transmit să poată detecta coliziunea.



Adaptorul A începe să transmită un cadru, și cu puțin timp înainte ca semnalul să ajungă la B, adaptorul B începe să transmită; astfel încât B a transmis doar un număr mic de biți înainte de abandon. Acești biți se vor propaga spre A, însă există posibilitatea să nu aibă destulă energie pentru ca A să detecteze o coliziune. Pentru ca A să detecteze coliziunea (și să renunțe), B transmite secvențe de bruieră.

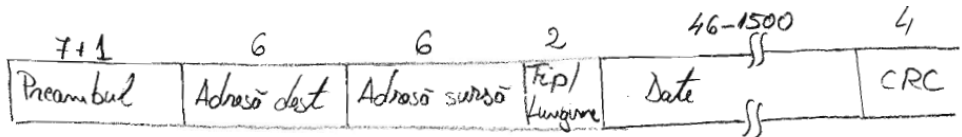
Algoritmul de regresie:

- coliziune 1: $\{0, 1\} \times 512 \times t_{\text{bit}}$
- coliziune 2: $\{0, 1, 2, 3\} \times 512 \times t_{\text{bit}}$
- coliziune 3: $\{0, 1, 2, 3, 4, 5, 6, 7\}$

coliziune ≥ 10 $\{0, 1, \dots, 1023\}$

Eficiența Ethernet-ului:

$$\eta = \frac{1}{1 + 5 \frac{t_{prop}}{t_{trans}}}$$

Structura cadrelor Ethernet:

- este folosit pentru sincronizare, primii 7 octeți: 10101010
 , ultimul octet: 10101011
- Adresa destinație: conține adresa adaptorului căruia îi este adresat cadrul. Dacă un adaptor recepționează un cadru cu o adresă destinație diferită de adresa proprie sau adresa de difuzare ignoră cadrul
- Adresa sursă: adresa adaptorului care a generat cadrul, comunicație bidirecțională
- Tip: - dacă valoarea este mai mică de 1536 indică lungime
 - dacă este mai mare de 1536 indică tipul
- Date: - acest câmp transportă (de regulă) un pachet IP. MTU pentru Ethernet este 1500 bytes, ceea ce înseamnă că lungimea pachetului nu trebuie să depășească această valoare. Dimensiunea minimă este de 46 bytes. Pachetele sub această valoare se completează până la 46 byte. Atunci când se utilizează completarea datele se transmit nivel superior cu tot cu completare
- CRC, CRC-32, se folosește pentru detecția erorilor. Cadrele recepționate cu CRC-ul incorect sunt ignorate

Coduri cu redundanță ciclică:

Cu toate că schema de mai sus poate fi uneori adecvată, în practică este larg utilizată o altă metodă: codul polinomial (cunoscut și sub numele de cod CRC, Cyclic Redundancy Code). Codurile polinomiale sunt bazate pe tratarea șirurilor de biți ca reprezentări de polinoame cu coeficienți 0 și 1. Un cadru de k biți este văzut ca o listă de coeficienți pentru un polinom cu k termeni, de la x^{k-1} la x^0 . Aritmetica polinomială este de tip modulo 2, nu există transport la adrese și nici împrumut la cadre, adunări și scăderi fiind identice cu XOR.

Atunci când este utilizată metoda cadrului polinomial, emițătorul și receptorul se pun de acord în avans asupra unei polinoame generator $G(x)$, la care atât bitul cel mai semnificativ cât și cel mai puțin semnificativ să fie 1.

Pentru a calcula suma de control pentru un cadru de m biți corespunzător polinomului $M(x)$, cadrul trebuie să fie mai lung decât polinomul generator. Ideea este de a adăuga

la sfârșitul cadrului o sumă de control astfel încât, polinomul rezultat să fie divizibil prin $G(x)$. Algoritmul de calcul este următorul:

1. Fie r gradul lui $G(x)$. Se adaugă r biți la capul mai puțin semnificativ al cadrului astfel încât acesta va avea $m+r$ biți și va corespunde lui $X^r M(x)$
2. Se împarte șirul de biți ce corespund lui $G(x)$ într-un șir de biți corespunzând lui $X^r M(x)$, utilizând împărțirea mod 2.
Practic se aliniază biții care reprezintă intrarea pe o linie, și se poziționează biții corespunzători polinomului generator sub marginea din stânga a liniei.
Dacă bitul de intrare aflat deasupra bitului cel mai din stânga al divizorului este 0 se deplasează divizorul la dreapta cu 1 bit, altfel se face XOR între divizor și intrare, după care divizorul este deplasat la dreapta, până atinge extrema.
3. Se scade restul din șirul de biți corespunzător lui $X^r M(x)$

Exemplu: 1101011011, $G(x)=x^4+x+1$, (10011)

$$\begin{array}{r}
 11010110110000 \\
 \underline{10011} \\
 01001110110000 \\
 \underline{10011} \\
 0000010110000 \\
 \underline{10011} \\
 00101000
 \end{array}
 \qquad
 \begin{array}{r}
 101000 \\
 \underline{10011} \\
 001110
 \end{array}$$

Structura adreselor:

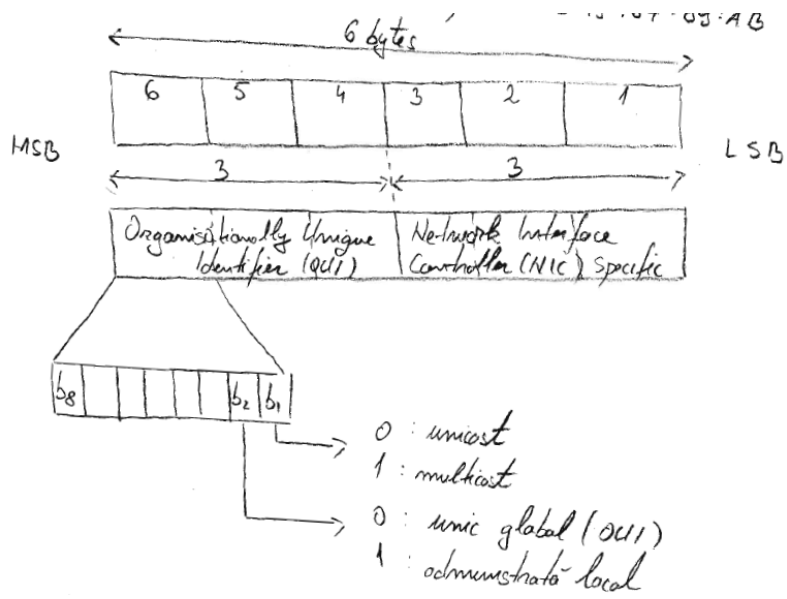
Adrese MAC, adrese fizice, adrese hardware

Adresele folosite la Ethernet sunt de tipul EUI-48 (Extended Unique Identifier),

48 biți = 6 octeți

Notăție (în baza 16, grupe de câte două cifre)

Exemplu: 01-23-45-67-89-AB, 01:23:45:67:89:AB



Adresa de difuzare: toți biții 1, FF-FF-FF-FF-FF-FF