

CAPITOLUL I
ASPECTE FUNDAMENTALE ALE SISTEMELOR
DIGITALE DE COMUNICAȚII

I.1 Introducere

Sistemele de comunicații, componente esențiale ale infrastructurii societății contemporane, cunosc în prezent o evoluție spectaculoasă datorată progreselor tehnologiilor în domeniul realizării circuitelor (VLSI - *Very Large Scale Integration*), a mediilor de transmisie și cele ale teoriei comunicațiilor. Deși inițial aceste sisteme s-au dezvoltat în variantă analogică, în prezent comunicațiile digitale iau amploare prin avantajele deosebite pe care le oferă.

Mediile de transmisie s-au diversificat: linii telefonice, fibre optice, unde radio, ghiduri de undă, discuri magnetice pentru înregistrări digitale (CD - *Compact Disk*) etc.

Gama aplicațiilor și numărul beneficiarilor sunt tot mai mari: transmisii digitale audio/video cu diferite nivele de calitate, televiziune de înaltă definiție (HDTV - *High Density TeleVision*), comunicații mobile, transmisii prin satelit, poșta electronică, rețele extinse de calculatoare, baze de date, tranzacții financiare computerizate, controlul proceselor industriale în timp real, învățământ la distanță, aplicații medicale, telemetrie etc.

Se dezvoltă sisteme și rețele digitale de comunicații de date de mare viteză, sisteme de înregistrare cu densitate mare a datelor pe suport, cu echipamente tot mai performante și costuri relativ reduse: procesoare digitale de semnal, procesoare de imagine de tip CNN (*Cellular Nelinear Networks*) [Chu97] care efectuează 1 TERA OPS (10^{12} *operations per seconds*), filtre de egalizare adaptive, circuite digitale de (de)codare și (de)modulare cu eficiență spectrală sau de putere, circuite de multiplexare bazate pe diferite tehnici de acces multiplu (TDMA - *Time Division Multiple-Access*; STDMA - *Statistical TDMA*; FDMA - *Frequency Division Multiple-Access*; CDMA - *Code Division Multiple-Access*), sisteme cu spectru extins (SSS - *Spread Spectrum Systems*) [Pet95].

Aspecte legate de sistemele biologice (sisteme fuzzy, rețele neuronale) permit dezvoltarea unor tehnici de codificare cu eficiență sporită destinate proceselor de transmisie a unor fluxuri mari de date și eliminarea suprasaturației informaționale a sistemelor de comunicații.

Comunicațiile de date de bandă largă (B-ISDN - *Broadband Integrated Digital Network*) [Jaj93] cu viteze de transmisie a informației de ordinul sutelor de megabiți pe secundă (Mbps) și mai mari sunt azi în atenția cercetătorilor. Rețelele de comunicații pe

arii locale cu viteze de ordinul Gbps (LAN) [Ger91] precum și cele optice sincrone (SONET) [Che92] permit lărgirea gamei aplicațiilor în domeniul comunicațiilor și al computerelor, punând totodată problema creșterii eficienței lor [Kun92], [Git92].

I.2 Structura sistemelor de comunicații digitale

Varianta cea mai simplă de realizare a comunicațiilor digitale o constituie sistemele de comunicații digitale (SCD) de tip "*punct cu punct*" [Bla90]. În contrast cu acestea, rețelele digitale de comunicație sunt de tip *multiacces* permițând accesul mai multor utilizatori la un canal de transmisie comun. *Protocoalele de transmisie* asigură controlul traficului semnalelor de informație, evitarea blocajelor (respectiv saturarea bufferelor), respectarea ierarhiilor de priorități în cadrul acestor rețele.

Schema bloc a unui SCD "punct cu punct" este prezentată în figura I.1.

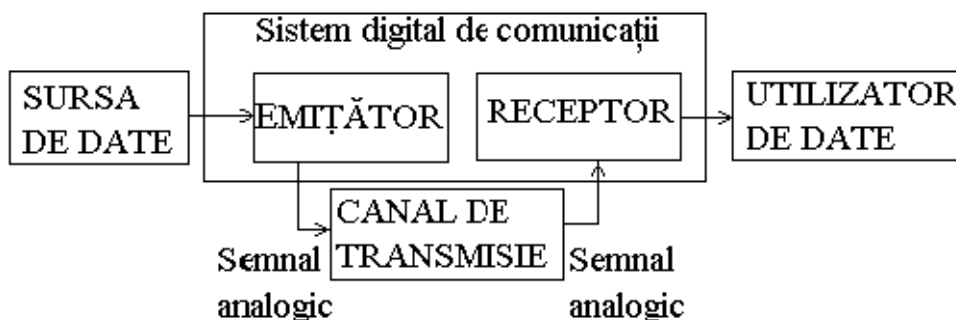


Fig. I.1 Shema-bloc a unui SCD

Sursa de date, canalul de transmisie și blocul utilizatorului nu sunt sub controlul proiectantului. Acesta proiectează doar emițătorul și receptorul, mai exact (de)codoarele și (de)modulatoarele digitale. Tendința actuală în proiectarea SCD este aceea de a separa proiectarea blocurilor aferente sursei și utilizatorului de cele ale canalului de transmisie (CT). Astfel sistemul are o flexibilitate crescută, ieșirea sursei fiind un semnal digital, o succesiune de biți, care poate fi transmis pe orice CT.

Funcțiile primare ale transmițătorului, referitoare la tehnicile de prelucrare a datelor, sunt ilustrate în figura I.2.



Fig. I.2 Schema-bloc a transmițătorului digital

◆ **Codorul sursei** generează un șir de biți compact în care se succed cuvintele de cod ale sursei. Prin operația de **compactare** se scade redundanța semnalului și debitul informațional. Dacă valoarea acestuia depășește capacitatea canalului de transmisie se poate realiza **compresia** datelor în limitele admise de aplicație, prin reducerea entropiei semnalului. La ieșirea acestui codor, natura sursei (audio, video, date) nu mai poate fi dedusă.

◆ **Blocul de cifrare** asigură securitatea transmisiei, prevenind preluarea și/sau falsificarea informațiilor de către un receptor/transmițător neautorizat.

◆ **Codorul de canal** crește redundanța semnalului transmis pentru a permite receptorului să realizeze detecția și corecția erorilor, respectiv adaptarea semnalului transmis la constrângerile impuse de CT. Simbolurile din semnalul de ieșire din codorul de canal (*channel codestream*) aparțin alfabetului canalului de transmisie care nu mai este neapărat binar.

◆ **Modulatorul digital** transformă secvența de simboluri discrete, în particular binare, într-un semnal analogic, prin asocierea fiecărui simbol de date cu unul analogic din alfabetul modulatorului. Semnalul modulat poate fi transmis direct pe CT.

◆ În cazul SCD de bandă largă (transmisii pe fibră optică sau prin satelit) se preferă extinderea prealabilă a spectrului semnalului transmis pentru protecția la zgomote și interferențe nedorite (cum ar fi semnalele de bruij) sau pentru multiplexarea semnalelor în cod (CDMA). Aceste sisteme sunt cunoscute sub numele de **sisteme cu spectru extins** (SSS - *Spread Spectrum Systems*).

Semnalul recepționat diferă de cel transmis datorită perturbațiilor aferente canalului de transmisie: zgomot aditiv Gaussian sau ne-Gaussian; fading; distorsiuni liniare (de amplitudine sau de fază) sau neliniare (ex: de intermodulație); interferențe.

Receptorul realizează operațiile inverse celor de la emisie (figura I.3).

◆ **Decodorul de canal** folosește redundanța cuvintelor de cod pentru detecția și corecția eventualelor erori. Acest fapt contribuie fie la creșterea calității transmisiei, fie

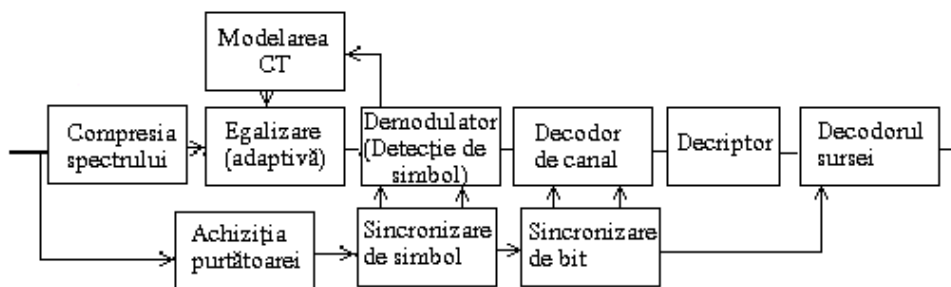


Fig.I.3 Funcțiile primare ale receptorului digital

la reducerea puterii emițătorului și a dimensiunilor antenelor de emisie/recepție, făcând posibilă funcționarea sistemului la rapoarte de puteri semnal/zgomot (SNR- *Signal-to-Noise Ratio*) mai mici, pentru o anumită valoare, impusă, a ratei erorilor în receptor.

◆ Modemurile actuale conțin **filtre egalizoare** cu parametri ajustabili, funcție de condițiile de transmisie de pe canal.

Tehnicile de modulație digitală constituie punctul forte al comunicațiilor digitale.

Îmbunătățirea performanțelor acestora se realizează prin asocierea cu metode specifice de codare a stărilor introduse de Ungerboeck [Ung77], [Ung82] (TCM - *Trellis Coded Modulation*) care au condus la apariția unor noi generații de modemuri.

Codurile aplicate semnalului de date se pot clasifica, din punctul de vedere al **structurii** lor, în trei mari categorii:

- ❶ coduri - bloc;
- ❷ coduri - prefix;
- ❸ coduri - arbore.

Codurile de tip bloc împart secvența de intrare în blocuri de k simboluri din alfabetul sursei, pe care le asociază cu secvențe de cod de n simboluri din alfabetul canalului.

Rata de codare se definește ca raport k/n .

Există coduri-bloc cu lungimi ale cuvintelor de intrare sau de ieșire fixe sau variabile:

- ◆ k fix - n fix;
- ◆ k fix - n variabil;
- ◆ k variabil - n fix;
- ◆ k variabil - n variabil (în particular, rata de codare poate fi constantă).

Codurile-prefix reprezintă un caz particular de cod-bloc cu lungime fixă a cuvintelor de intrare și variabilă la ieșire cu proprietatea esențială că nici o combinație codată nu este prefix pentru alt cuvânt de cod (așa-numita **condiție de prefix**). Astfel se asigură **autosincronizarea** semnalului codat adică orice șir de simboluri codat se împarte în mod unic în cuvinte de cod.

Exemplu: Fie o sursă informațională generatoare de 3 caractere: (A, B, C) care se asociază cu setul de cuvinte de cod: (00; 01; 1).

Secvența de intrare ABCBABBC se codează: 00011010001011.

În receptor biții se citesc individual iar lungimile secvențelor de cod se contorizează. Întrucât primul bit 0 nu are echivalent în tabelul de codare, se va citi următorul bit și se va căuta simbolul asociat combinației 00 (lungimea contorizată în acest moment este 2). Se identifică A după care se resetează contorul și se caută următoarea combinație de cod.

Codurile de tip arbore codează fiecare cadru de intrare printr-o secvență de cod în funcție de **m** cadre anterioare și **a** cadre ulterioare. Parametrii **m** și **a** reprezintă memoria respectiv anticipația codului. Codarea se va face pe baza unei diagrame de tip arbore. În practică se utilizează codurile cu număr finit de stări, adică cele cu memorie și anticipație finite. Codurile-arbore invariante în timp se numesc **coduri-trellis** întrucât funcționarea lor poate fi urmărită pe baza unei diagrame de tip grilă (*trellis*). Codurile-trellis liniare sunt denumite **coduri convoluționale**.

Observație: Transmisia semnalelor analogice (voce; audio, video) prin sistemele de comunicații digitale este posibilă doar dacă, în prealabil, se realizează discretizarea lor. Procesul de discretizare constă în operații succesive de filtrare, eșantionare, cuantizare și codare.

Din punctul de vedere al funcției îndeplinite, tehnicile de codare a datelor se împart în următoarele categorii:

❶ **coduri de compactare**, care reduc redundanța semnalului digital fără a afecta entropia acestuia;

❷ **coduri de compresie** care reduc entropia semnalului în scopul micșorării debitului binar al sursei informaționale;

❸ **coduri de criptografiere** a datelor pentru asigurarea secretului transmisiei;

❹ **coduri corectoare de erori** care prin creșterea redundanței semnalului transmis fac posibile detectia și corecția unui anumit număr de erori de transmisie;

❺ **coduri de translare** a datelor în vederea adaptării acestora la constrângerile canalului de comunicații.

I.3 Generalități privind rețelele de comunicații

Rețelele de radiorelee de transmisie în teritoriul a programelor de televiziune sau a celor de radiodifuziune precum și rețeaua telefonică fixă comutată s-au dezvoltat inițial în variantă analogică.

Rețelele de comunicații au devenit importante relativ recent prin dezvoltarea explozivă a industriei producătoare de calculatoare și echipamente asociate (imprimante, scanere, fax-modemuri etc). Tendința actuală în proiectarea rețelelor de comunicație este de utilizare a circuitelor digitale în tehnologie VLSI.

În formă modernă, rețelele digitale de comunicații sunt de tip **multiacces**, adică mai mulți utilizatori sunt conectați la un singur canal de transmisie. Regulile respectate de utilizatori într-o rețea digitală de comunicații se numesc **protocoale**.

I.3.1 Entități

Comunicația se realizează între două echipamente denumite generic **entități**. Transferul informației se poate face pe un **canal de transmisie** (adesea denumit restrictiv **linie**), de natură foarte diversă: aer (eter), conductor metalic, fibră optică.

La un anumit moment, o entitate poate fi:

◆ **transmițător**, dacă transmite semnal;

◆ **receptor**, dacă primește informații;

◆ **transmițător și receptor**, dacă transmite și recepționează informații în același timp.

Aceste tipuri de entități corespund celor trei *moduri specifice de transmisie* pe un anumit canal de comunicații:

◆ **simplex** - informația circulă într-un singur sens, de la *transmițător* la *receptor* (exemplu: introducerea datelor de la tastatura unui calculator).

◆ **semi-duplex** - ambele entități care comunică între ele pot transmite informații, dar nu simultan. Când rolurile se inversează, este necesară schimbarea sensului de transmisie pe linie (*line turnaround*) și deci un timp suplimentar neutilizat pentru transferul de informație (de exemplu, în cazul comunicațiilor prin stații radio).

◆ **duplex** - ambele entități transmit și recepționează informații în același timp, iar



Fig.I.4.(a) Transmisie simplex

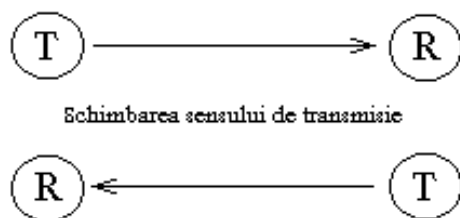


Fig.I.4. (b) Transmisie semi-duplex

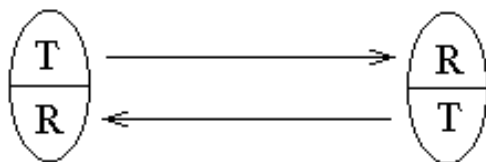


Fig.I.4. (c) Transmisie duplex

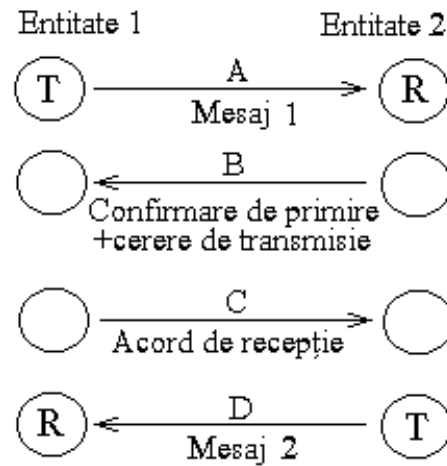


Fig.I.4. (d) Schimbarea sensului de transmisie pe linie în modul semi-duplex

canalul de comunicații realizează transmisia fără interferențe, de exemplu prin separarea benzilor de frecvențe utilizate pe cele două sensuri (transmisii pe linii telefonice pe patru fire).

Observație: Modul semiduplex de transmisie are avantajul utilizării întregii benzi de frecvențe disponibile pe canal la un preț rezonabil comparativ cu modul duplex dar și dezavantajul că necesită un sistem de control (software) mai complex precum și timpi suplimentari pentru schimbarea sensului de transmisie pe linie (Fig.I.4).

I.3.2 Echipamentul de transmisie a datelor (DTE)

În cel mai simplist mod, o entitate poate fi privită ca un dispozitiv care având specificată *intrarea*, o *procesează* și generează o anumită secvență de *ieșire*. În cazul în

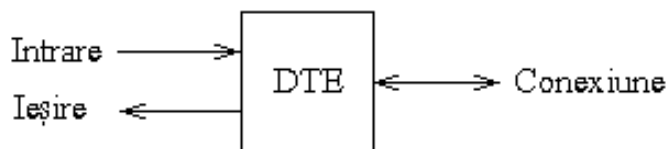


Fig.I.5 Reprezentarea simbolică a DTE

care acesta comunică cu o altă entitate (computer, imprimantă etc), este necesară stabilirea tipului de conector folosit, computerul fiind privit ca un **echipament terminal de date** (DTE - *Data Terminal Equipment*). În figura I.5 este reprezentat simbolic un DTE.

Pentru a realiza transmisia este necesar ca DTE-ul să fie asociat cu un **echipament pentru comunicații de date** (DCE - *Data Circuit-Terminating Equipment*) care adesea este un **modem**.

I.3.3 Protocoale

Comunicația între două entități (computere sau procese) este controlată în ceea ce privește starea fiecărei părți implicate, condițiile sau modul de transmisie, prin așa-numitele **protocoale de comunicații**.

Un protocol de transmisie stabilește:

- ◆ modul de *reprezentare* (codare sau semnalizare) a informației;
- ◆ modalitatea de *control* a traficului informațional pe canal;
- ◆ *codul detector și corector de erori* aplicat;
- ◆ *eficiența* de utilizare a canalului de transmisie;
- ◆ metoda folosită pentru *sincronizarea* entităților;
- ◆ gradul de *transparentă* a transmisiei, astfel încât utilizatorii să nu se confrunte

direct cu anumite detalii de transmisie.

Protocoalele pentru comunicațiile de date operează pe nivelele logice de transmisie deci deasupra celui fizic (conform modelului de referință OSI) care presupune cunoașterea nivelelor sau a tensiunilor de semnal. Pe nivelul logic inferior, se utilizează **protocoale de date orientate pe structura de biți** (BOP - *Bit-Oriented Protocol*) sau **pe structura de caractere** (BCP - *Byte Controlled Protocol*).

A. Codarea informației

Nu există o metodă universală, standard, de codare a informației dar producătorii de echipamente pentru comunicațiile digitale au convenit asupra formatului de bază,

binar, cu biți "1" și "0". Informația poate fi reprezentată binar sau M-ar (octal; hexazecimal etc).

Terminologia specifică transmisiilor de date include, pentru reprezentarea binară cu biți 1 și 0, alte exprimări echivalente precum: ON/OFF, Low-High, True-False, Mark-Space.

Biții pot fi grupați câte 4: □□□□ (*nibble*) sau câte 8: □□□□□□□□ (byte sau octet).

Descrierea, respectiv citirea, acestor cuvinte este dictată de producătorul echipamentului. De exemplu, numerotarea biților într-un octet poate fi făcută de la 1 la 8 sau de la 0 la 7, de la stânga la dreapta sau de la dreapta la stânga, crescător sau descrescător. Valoarea unui octet depinde de asemenea de convenția adoptată de producător. Dacă se lucrează fără semn (*unsigned*), octetul poate lua orice valoare de la 0 la 255. În caz contrar (*signed bytes*), domeniul de valori va fi de la -128 la +127.

Codarea informației presupune deseori *codarea caracterelor*.

Cele mai utilizate coduri construite pe structura de caractere sunt:

◆ **EBCDIC** (*Extended Binary Coded Decimal Interchange Code*) cod pe 8 biți, proiectat de firma IBM (International Business Machines Corporation) pentru computerele anilor '60;

◆ **ASCII** (*American Standard Code for Information Interchange*), cod pe 7 biți, folosit de majoritatea firmelor producătoare de tehnică de calcul, inclusiv de IBM.

Sunt codate caractere grafice (printabile) precum și unele de control.

În aplicații se lucrează fie cu numere întregi, respectiv *cuvinte* de 16 biți (2 octeți) sau 32 biți (4 octeți), fie cu numere fracționare (cu mantisă și exponent: *floating-point number*), cu virgulă fixă sau mobilă, exprimabile pe 32, 64 sau 128 biți.

Numerele întregi, exprimate în format BCD (Binary Coded Decimal), pot fi reprezentate în diferite variante:

◆ *'zoned decimal'*:

cifrele : exprimate binar de la 0000 la 1001, în nibble-ul din stânga fiecărui octet, subliniat cu o linie;

semnul : pozitiv 1100, negativ 1101; număr pozitiv, fără semn: 1111; plasat în nibble-ul drept al ultimului octet.

De exemplu:

+794: 1111 0111 1111 1001 1100 0100

-325: 1111 0011 1111 0010 1101 0101

860: 1111 1000 1111 0100 1111 0000.

◆ 'packed decimal':

Se codează în binar cifrele numărului iar semnul este plasat la final.

De exemplu:

+794: 0111 1001 0100 1100;

-325: 0011 0010 0101 1101;

860: 1000 0100 0000 1111.

Pentru a realiza comunicarea între două entități este necesar acordul lor asupra modului unic de reprezentare a datelor. Sunt necesare deci convertoare de tipul ASCII - EBCDIC. Probleme tot apar întrucât modul de stocare a informației diferă de la un producător la altul. De exemplu, în procesorul INTEL 80x86, un șir de bytes ABCDEF este stocat ca în figură:

15.....	8	7	0
B	A		
D	C		
F	E		

în timp ce același șir va fi ordonat în memorie de un procesor MOTOROLA 68000 după cum urmează:

15.....	8	7	0
A	B		
C	D		
E	F		

Astfel o transmisie de date pe 8 biți în format ASCII, între un procesor Intel și unul Motorola, se va realiza corect. Dar într-o transmisie pe 16 biți digiții din fiecare cuvânt vor fi inversați (se va citi BA în loc de AB etc).

B. Controlul transmisiei informației

Se utilizează diferite secvențe de control în procesul comunicației:

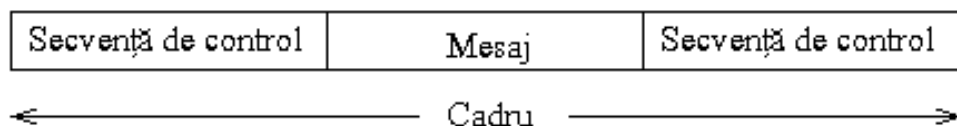


Fig.I.6 "Împachetarea" mesajului informațional cu secvențele de control

- ◆ adresa destinatarului;
- ◆ acordul de recepție;
- ◆ secvențele de testare a condițiilor de transmisie;
- ◆ secvențele de cod pentru inversarea sensului de transmisie pe un canal semiduplex;
- ◆ secvențele de separare a cadrelor informaționale (STX - *Start Text*; EOT - *End Of Text*);
- ◆ secvențe de confirmare (ACK- *ACKnowledge*) sau infirmare (NAK - *Not Acknowledge*) a primirii semnalului informațional fără erori ș.a.

Toate acestea nu transportă propriu-zis informație dar asigură continuitatea transmisiei și recepția fără erori a datelor. Informația adițională necesară bunei desfășurări a comunicației, așa-numita **informație de control**, se transmite fie înainte, fie după mesajul propriu-zis, ca antet (*overhead*) sau ca secvență finală (*trailer*), formând *cadrele* sau *pachetele* de date conform unui standard agreat de cele două entități implicate în transmisie (Fig.I.6).

Secvențele de control sunt stabilite prin protocolul de transmisie acceptat. Apare evident problema recunoașterii secvențelor de control în interiorul semnalului transmis pentru a nu se confunda cu secvențele de date și pentru evitarea preluării unor comenzi false. De exemplu, dacă secvența de date 0000100 este considerat caracter de control EOT în cod ASCII, atunci receptorul nu va mai citi restul șirului informațional transmis.

C. Detectia și corecția erorilor

Pe canalul de comunicație apar diferite erori datorate unor perturbații precum:

- ◆ *atenuarea* de semnal proprie canalului de comunicație, funcție de distanța dintre transmițător și receptor și de natura mediului de transmisie (exemplu: 4dB/km în

cazul transmisiei prin conductor metalic; 0,004 dB/km pentru transmisie pe fibră optică realizată din materiale pe bază de fluoruri);

◆ *distorsiunile liniare de amplitudine și fază* introduse de canal în semnalul transmis (posibil a fi corectate prin filtre adaptive de egalizare amplasate în receptor);

◆ *distorsiunile neliniare*, de exemplu distorsiunile de intermodulație, care determină apariția unor armonici noi în spectrul semnalului recepționat comparativ cu cel transmis. (În acest sens, tehnicile digitale de modulație cu anvelopă constantă sunt avantajoase întrucât pot fi folosite și în sisteme de transmisie cu amplificatoare de putere neliniare lucrând în regim de saturație.);

◆ *interferențe nedorite cu zgomot aditiv* de natură termică, *alb* sau *colorat* (cu caracteristică de putere constantă sau variabilă în banda utilă), cu distribuție probabilistică *Gauss* sau *Rayleigh* (influența acestui zgomot poate fi minimizată prin introducerea unor filtre adaptate special proiectate pentru raport de puteri semnal/zgomot maxim);

◆ *interferențe cu canalele adiacente*, în frecvență sau spațial (fenomenul de ecou), *cu canalul în cuadratură* (în demodolatoarele digitale lucrând cu semnale purtătoare ortogonale):

ACI - *Adjacent Channel Interference*;

CCI - *CoChannel Interference*;

QCI - *Quadrature Channel Interference*;

◆ *interferențe de natură electromagnetică* (zgomote de tip impuls) cauzate de fenomene naturale sau cuplaje cu liniile de înaltă tensiune;

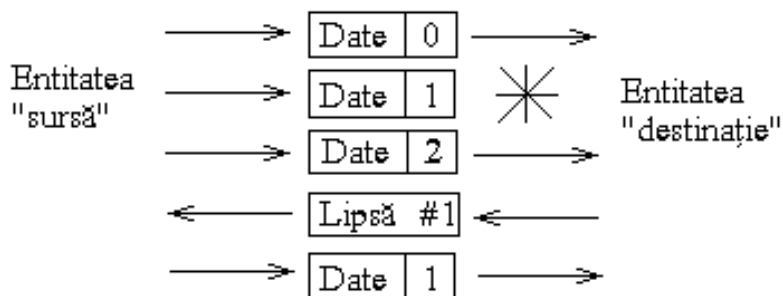


Fig.I.7 Retransmisia unui mesaj "pierdut"

◆ fenomenul de *fading* (atenuarea puternică a semnalului transmis în intervale de timp aleator distribuite în timp, cu durate variabile) în cazul transmisiei prin undă radio (exemplu: comunicații mobile), determinat de suprapunerea semnalului transmis cu unele replici ale acestuia, întârziate și defazate (chiar în opoziție de fază), reflectate de obstacole fixe (clădiri, forme de relief) sau mobile (alte vehicule);

◆ *întreruperea transmisiei* din cauza distrugerii mediului fizic de transmisie, ceea ce implică retransmisia unor cadre după restabilirea legăturii.

Este necesară deci transmisia unor secvențe de control care să ateste recepția corectă a datelor sau să semnaleze erorile de transmisie (Fig.I.7).

Pentru corecția erorilor pot fi aplicate diferite tehnici de codare (Hamming; Reed-Solomon; Coduri ciclice - CRC - *Cyclic Redundancy Checking* cu diferite polinoame generatoare prime de ordin 12; 16; 32 stabilite de CCITT etc).

În echipamentele mai simple, respectiv mai ieftine, se detectează erorile fără a se face corecția acestora, urmând a se cere retransmisia cadrelor afectate de erori (ARQ - *Automatic Repeat under Request*) prin diferite procedee:

- ◆ reluarea ultimelor n cadre (*go-back-n*);
- ◆ reluarea selectivă a cadrelor eronate (*selective repeat*);

Există protocoale care permit transmisia cadrului curent doar după confirmarea recepției corecte a celui anterior (*stop-and-wait*). Astfel se elimină timpii de retransmisie a anumitor cadre corect recepționate precedate de unul eronat.

D. Eficiența canalului de comunicație

Informația poate fi transferată de la sursă la destinatar prin diferite medii de transmisie. Transferul respectiv poate fi caracterizat prin timpul și viteza de transmisie, rata de erori, ponderea secvențelor de control în semnalul transmis, fluența traficului (retransmisii, congestii etc), modul de transmisie (serie sau paralel). În mod ideal, eficiența comunicației poate fi privită prin prisma vitezei de transmisie, respectiv a cantității de informație transferată într-un anumit timp, ceea ce ar presupune reducerea numărului de biți de control transmiși, adică o scădere a calității transmisiei. Este necesar deci un compromis viteză - calitate.

Pe de altă parte, transmisia paralelă a datelor (de exemplu, pe 8 biți) conduce la o creștere a vitezei de transmisie dar și a prețului canalului de comunicație, comparativ

cu modul serial de transfer a datelor. Este necesar deci și un compromis preț - viteză de transmisie.

În general, transmisia în paralel a datelor se realizează pe distanțe relativ scurte între entități (de exemplu, într-un birou, între calculator și imprimantă, la cel mult 15 m) astfel încât efectele interferențelor să fie minime. În cazul transmisiei datelor pe distanțe mari, se vor utiliza canale de comunicație seriale.

E. Sincronizarea în sistemele de comunicații

Protocoalele de comunicații obligă entitățile implicate într-o transmisie să lucreze sincron (*in-step*), adică pe baza aceluiași semnal de tact. Într-un calculator, apar diferite semnale de sincronizare funcție de nivelul pe care se lucrează. Pe nivelele inferioare este necesară sincronizarea de bit, byte sau cadru. Întrucât entitățile lucrează independent una de cealaltă, transmisia între acestea se va realiza *sincron* prin intermediul unui semnal comun de sincronizare sau *asincron* dacă fiecare mesaj va fi precedat de informația de sincronizare.

Într-o transmisie *sincronă* șirul de date se împarte în blocuri cu lungime standard (uzual 256 caractere) care sunt încadrate pentru transmisie de caracterele de control: STX (start-of-text), ETX(end-of-text), de cele de sincronizare SYN (synchronize) (Fig.I.8). Modul sincron este util în transmisiile de date de mare viteză, pentru volume mari de date (exemplu: în rețelele de comunicații) fiind mai rar întâlnit în aplicațiile tehnice care implică utilizarea unui microcomputer.

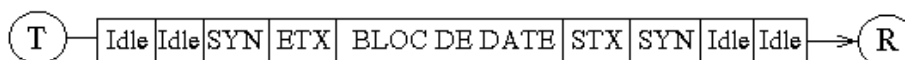


Fig.I.8 Bloc de caractere într-o transmisie sincronă

În anumite situații, semnalul de sincronizare poate fi utilizat pentru menținerea unei legături deja stabilite, pe intervalele în care nu se transmit mesaje. În acest scop, numeroase protocoale de comunicații prevăd transmisia periodică a unor cadre cu format

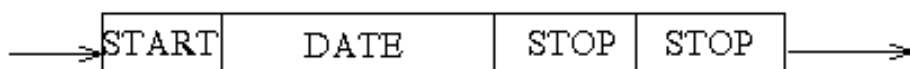


Fig.I.9 Bloc de caractere transmis asincron

Modul în care stațiile accesează canalul de comunicație depinde de felul în care se face selecția apelurilor (*polling*). Există două metode:

◆ **apelare individuală**, periodică a fiecărei stații, în ordinea adreselor, cu răspuns direct către server (*roll-call polling*) (Fig.I.11);

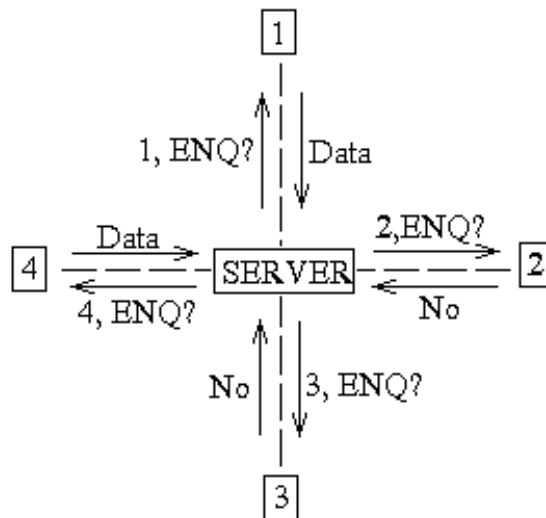


Fig.I.11 Exemplu de apelare de tipul *roll-call*

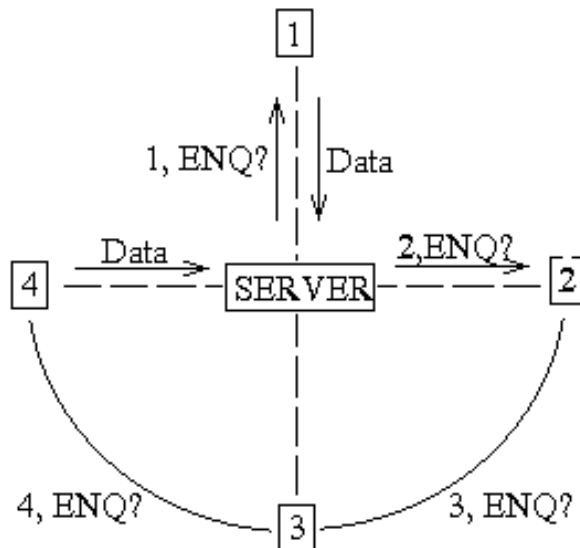


Fig.I.12 Exemplu de apelare de tip *hub*

◆ **apelare recursivă** dinspre server sau dinspre stația precedentă, în situația în care aceasta nu are date de transmis la acel moment (*hub go-ahead polling*) (Fig.I.12).

Astfel se reduc anumiți timpi de apel sau răspuns. Există însă pericolul pierderii apelului de la o stație către alta și blocarea totală a transmisiei.

Observații: 1. ENQ = ENQUIRE (cerere de transmisie).

2. Stațiile subordonate pot fi ierarhizate ca priorități și apelarea acestora se va face funcție de acestea.

H. Exemple de protocoale (Bisync, SDLC, ACK/NAK, XON, XOFF)

Protocolul BISYNC, creat de firma IBM, *Binary Synchronous* (abreviat Bisync sau BSC) este un protocol cu control pe byte (BCP) și are diferite formate de control. În figura I.13 sunt prezentate două structuri ale protocolului BSC, una de transmisie a datelor și a secvenței de control (*header*) cealaltă numai a adresei.

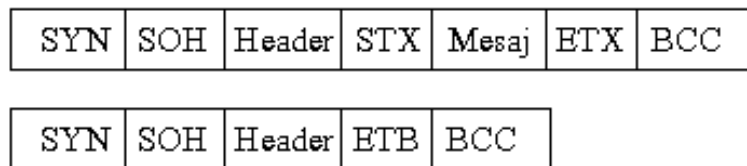


Fig.I.13 Structuri ale protocolului BSC

Notatii:

SYN	Synchronize	Sincronizarea transmițătorului și receptorului.
SOH	Start of Header	Avertizează receptorul că până la STX se transmite secvența de control.
Header		Identifică receptorul și conține informații de control.
STX	Start of Text	Atenționează receptorul că până la ETX se transmite un mesaj.
Text		Mesaj
ETX	End of Text	Mesajul este transmis complet.
ETB	End of Transmission Block	Se utilizează numai pentru header.

EOT	End of Transmission	Apare după ultimul mesaj.
BCC	Block Check Character	Secvență de cod transmisă pentru detecția și corecția erorilor.

Pentru a nu crea constrângeri asupra mesajelor, trebuie separată secvența de control de cea de date. Se încadrează secvența de date cu caracterul de control DLE (Data Link Escape) astfel încât transmisia să fie **transparentă**. Combinația DLE STX avertizează receptorul să ignore orice caracter de control până la apariția secvenței compuse ETX DLE. Probabilitatea apariției unei secvențe DLE în șirul de date este însă nenulă iar micșorarea ei se face prin dublarea oricărei secvențe de date de forma DLE (*character stuffing*) deci se va transmite DLE DLE urmând ca receptorul să ignore unul din aceste două blocuri. Protocolul Bisync poate fi utilizat doar în transmisii semiduplex, fiind necesară confirmarea/infirmarea recepției fiecărui câmp (header sau mesaj) cu secvențele ACK (*ACKnowledge*) sau NAK(*Not AcKnowledge*). Există protocoale de tip BCP care lucrează în cod ASCII și care pot fi utilizate și în transmisii duplex. Protocoalele orientate pe octeți se utilizează în rețele de mari dimensiuni precum și pentru accesarea unor baze mari de date.

Protocolul SDLC

Protocoalele cu orientare pe bit (BOP) au câștigat mult în popularitate în ultimii ani în defavoarea celor BCP. Firma IBM a creat protocolul SDLC (*Synchronous Data Link Control*) care elimină orice secvență de control, acesta realizându-se pe criteriul poziției biților dintr-un cadru.

Șirul de date este împărțit în *cadre* iar acestea, la rândul lor, în *câmpuri* cu anumite funcții (Fig.I.14):

◆ **Câmp de delimitare** (*Flag Field*), plasat la începutul și la sfârșitul fiecărui cadru SDLC;

Flag	Câmp de adresă	Câmp de control	Câmp informațional	Câmp de control a erorilor	Flag
------	----------------	-----------------	--------------------	----------------------------	------

Fig.I.14 Cadru SDLC

- ◆ **Câmp de adresă** specifică adresa entității apelate;
- ◆ **Câmp de control** specifică tipul transmisiei (apel sau răspuns) și conține informații despre pozițiile biților de control;
- ◆ **Câmp de informații**, cu lungime variabilă, conține mesajul codat ASCII sau EBCDIC;
- ◆ **Câmp de control a erorilor** conține informații pentru depistarea erorilor de transmisie.

Toate câmpurile, cu excepția celui informațional, sunt de un octet.

Acest protocol nu prezintă moduri distincte de transmisie și nici caractere speciale de control, fiind mult mai flexibil decât protocolul Bisync. În plus, are o capacitate mai mare de detecție a erorilor (în toate cadrele, pe toți biții de date și de control).

Se utilizează pentru câmpul de delimitare (*flag*) octetul 01111110 impunându-se constrângerea ca în câmpul informațional să nu apară secvențe de mai mult de 5 biți '1' consecutivi. Aceasta se rezolvă prin intercalarea câte unui '0' (*bit stuffing*) după fiecare 5 biți '1' din câmpul de date urmând ca receptorul să rejeteze biții suplimentari.

Din protocolul SDLC a fost dezvoltat protocolul HDLC (*High-level Data Link Control*) cu același format de cadru dar cu mici diferențe privind lungimea câmpului de delimitare (3 octeți) și codul corector de erori aplicat.

Protocoale de comunicare cu imprimanta

În cazul comunicației dintre un calculator și o imprimantă, interfețele respective vor funcționa pe baza protocolului de transmisie adoptat care va specifica modalitatea de control a intrării și ieșirii datelor (I/O).

Protocolul ACK/NAK atribuie calculatorului controlul transmisiei (Fig. I.15) în timp ce **protocolul XON/XOFF** permite imprimantei să dirijeze comunicația cu computerul (Fig. I.16).

Observație: XON = Transmit ON; XOFF = Transmit OFF. Caracterele DC1/DC3 (din grupul de caractere ASCII) îndeplinesc aceleași funcții ca și XON/XOFF.

Se observă că protocolul ACK/NAK este de tipul 'go-back-n' și are dezavantajul pierderii timpului inițial de transmisie a grupului de n cadre conținând unul sau mai

multe erori. Protocolul XON/XOFF funcționează pe principiul 'stop-and-wait' ceea ce conduce la creșterea timpului necesar pentru transmitia unui anumit volum de date. Avantajoasă ar fi, ca timp total de transmisie, utilizarea unui protocol cu retransmisie selectivă a cadrelor recepționate eronat.

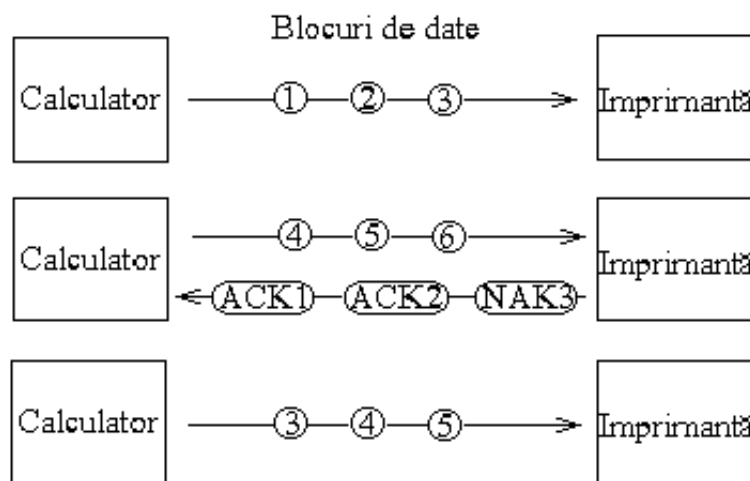


Fig.I.15 Transmisie cu protocolul ACK/NAK

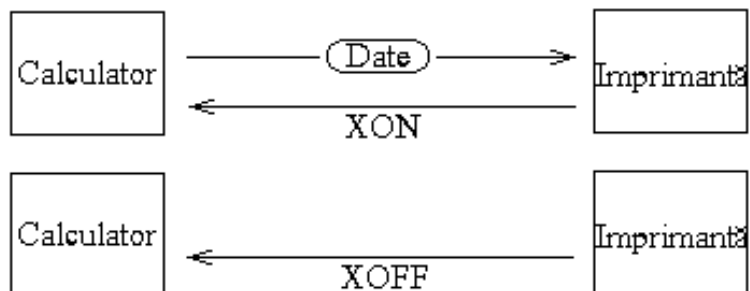


Fig.I.16 Transmisie cu protocolul XON/XOFF

Indicație: Se recomandă ca bibliografie pentru cunoașterea protocoalelor de comunicații, sincrone și asincrone, pentru cele 7 nivele ale modelului OSI, volumele "Computer Networks" de Andrew Tanenbaum [Tan88] și "Comunicații de date" de Miranda Naforniță ș.a. [Naf97].

I.3.4 Algoritmi ATM

Modul de transfer asincron a datelor (abreviat ATM - *Asynchronous Transfer Mode*) presupune aplicarea unor protocoale speciale de transmisie.

În rețelele de comunicații actuale, repartizarea intervalelor de transmisie (ca timp sau ca bandă de frecvență) se poate face:

◆ **cu alocare fixă**, prin multiplexarea canalelor: în timp (TDMA - *Time Division Multiple Access* sau STDMA - *Statistical TDMA*), în frecvență (FDMA - *Frequency Division Multiple Access*) sau în cod (CDMA - *Code Division Multiple Access*);

◆ **cu alocare la cerere.**

În primul caz, canalul de comunicații este ineficient utilizat dacă anumiți utilizatori nu au date de transmis ceea ce conduce la ne folosirea anumitor intervale de timp sau de frecvență (*time slot* sau *frequency slot*).

În cazul alocării la cerere eficiența de utilizare a canalului este maximă (100%) adică indiferent de numărul utilizatorilor care au date de transmis la un anumit moment întreaga bandă de frecvențe disponibilă este ocupată. Evident sunt necesari timpi suplimentari de aplicare ai unui protocol care să gestioneze canalul de comunicație, respectiv de planificare a transmisiilor. Există așa-numitele procesoare generale de planificare (GPS - *Generalized Processor Sharing*) care lucrează la viteze mari pentru reactualizarea permanentă a listelor de adrese ale utilizatorilor, respectiv pentru rescalarea lățimii benzilor alocate și a vitezelor de transmisie funcție de priorități, cu timpi de decizie relativ mici comparativ cu timpul efectiv utilizat pentru transferul de date.

Se pot exemplifica cele de mai sus prin protocolul de transmisie asincronă WFQ (*Weighted Fair Queing*) care asigură rate de transmisie variabile funcție de numărul cererilor și gradele de priorități.

Să considerăm transmisia cu alocare la cerere și multiplexare în frecvență într-o rețea cu trei utilizatori (A;B;C) și vector de priorități (2;1;1), pe un canal de comunicații binar, cu bandă de 40MHz.

Dacă transmit simultan toți utilizatorii, atunci repartizarea benzii canalului și a vitezelor de transmisie, se va face proporțional cu gradele de priorități adică:

A: 50% 20 MHz 24Mbps.

B: 25% 10 MHz 12Mbps.

C: 25% 10 MHz 12Mbps.

Se va calcula în fiecare interval de transmisie așa-numitul **timp virtual** care va stabili **valoarea de scalare** a benzii inițiale alocate.

Dacă la momentul inițial (0 secunde) A face cerere de transmisie iar B și C sunt inactivi, atunci ponderea calculată este 2 iar A dispune de întreaga bandă a canalului și transmite cu viteza maximă:

$$w_1 = \frac{1}{0.5} = 2 \rightarrow 40\text{MHz} \dots\dots 48\text{Mbps}.$$

Dacă la $2\mu\text{s}$ B face cerere de transmisie atunci banda canalului se împarte între cei doi utilizatori activi din acel moment cu ponderea:

$$w_2 = \frac{1}{0.5+0.25} = 1.33$$

Rezultă că A dispune de un canal de frecvență cu lățimea 26,66 MHz și viteza 32Mbps iar B folosește restul benzii, respectiv 13,34 MHz cu 16 Mbps.

La $5\mu\text{s}$ intervine cererea de transmisie a lui C adică din acest moment toți utilizatorii sunt activi și valoarea de scalare este 1, lățimile benzilor de frecvență alocate fiind cele calculate inițial.

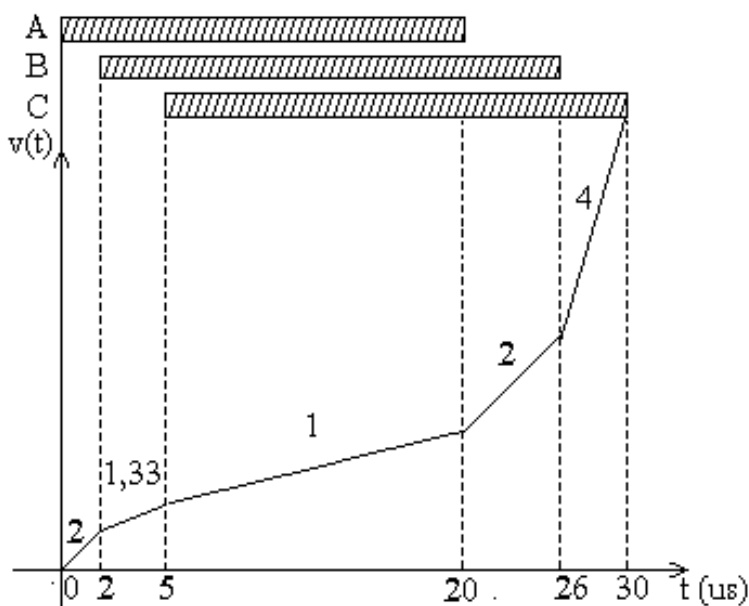


Fig.I.17 Timpul virtual de transmisie

La $20\mu\text{s}$, A încheie transmisia. Se determină valoarea de scalare pentru utilizatorii B și C:

$$w_4 = \frac{1}{0.25+0.25} = 2$$

Deci B și C transmit fiecare pe 20MHz cu 24Mbps.

La $26\mu\text{s}$, B termină de transmis datele astfel că în sistem rămâne activ doar C.

$$\text{Ponderea acestuia va fi: } w_5 = \frac{1}{0.25} = 4 .$$

Astfel C va dispune de întreaga bandă de 40 MHz și de viteza maximă de transmisie de 48 Mbps.

Exemplul considerat ilustrează o modalitate de maximizare a eficienței de utilizare a canalului de comunicații. În Fig. I.17 este reprezentat **timpul virtual de transmisie**, panta caracteristicii fiind proporțională cu valoarea de scalare pe fiecare subinterval. Se poate realiza și o reprezentare grafică în trepte a valorilor.

Se observă că timpul necesar transferului datelor de la un utilizator depinde de volumul mesajului, de cererile de transmisie prin rețea și de ierarhiile de priorități. Se obține astfel o caracteristică dinamică a volumului de date transferate până la un anumit moment, cu pantă de valoare variabilă.

În general, protocoalele pentru comunicații ATM (WFQ, FBQ - *Frame Based Queing* ș.a.) au ca scop:

- ◆ evitarea congestiilor;
- ◆ creșterea eficienței de utilizare a canalului.;
- ◆ respectarea priorităților;
- ◆ simplificarea implementării;
- ◆ micșorarea întârzierilor de transmisie (*end-to-end delay*) care pot ajunge la

valori relativ mari comparativ cu durata transmisiei (250 ms - pentru comunicații prin satelit).

I.3.5 Aplicații

P1. Pe un canal de comunicații trebuie transmis cadrul:

DLE, STX, A, DLE, B, DLE, ETX.

Dacă se aplică algoritmul '*character stuffing*', care va fi secvența de caractere transmisă?

P2. Dacă șirul binar 01111011111011111001111110 este transmis prin protocolul SDLC cu constrângerea de tip '*bit stuffing*', care va fi secvența decodată?

P3. Să se determine secvențele de control a erorilor (CRC) pentru șirul de date:

1101 0011 1000 1011 0001 1001

dacă se utilizează polinoamele:

$CRC-12 = x^{12} + x^{11} + x^3 + x^2 + x + 1$ cu lungime a blocului de date de 6 biți.

$CRC-16 = x^{16} + x^{15} + x^2 + 1$ cu lungime a blocului de date de 8 biți.

$CRC-CCITT = x^{16} + x^{12} + x^5 + 1$ cu lungime a blocului de date de 8 biți.

Indicație: Se va calcula restul împărțirii polinomului asociat șirului de date la cel generator. Operațiile între coeficienți se efectuează modulo-2.

P4. Se transmite pe linia telefonică în banda vocală, un fișier de date de 1,44 MB, cu viteză constantă de 4800 bps. Dacă se aplică protocolul SDLC cu lungimea cadrului informațional de 8 octeți (fără constrângeri de tip '*bit stuffing*'), care este durata totală de transmisie?

P5. Care este ordinea apelării utilizatorilor într-o rețea de comunicații de tip *roll-call*, cu trei utilizatori A, B și C, având gradele de priorități 2; 1; 1?

P6. Calculați și reprezentați grafic timpul virtual de transmisie pentru o comunicație în mod asincron cu protocol WFQ, pe un canal de bandă largă (48 kHz) și viteză de transmisie a datelor de 56 kbps, la care sunt conectați 6 utilizatori A; B; C; D; E și F, cu ordinea de priorități: 2; 2; 1; 1; 1; 1. Intervalele de transmisie sunt ocupate după cum urmează:

0 - 10 ms: A activ;

10 ms - 25 ms: A și C activi;

25 ms - 40 ms: B, C și D activi;

40 ms - 60 ms: B, C, D și E activi;

60 ms - 75 ms: C, D, E și F activi;

75 ms - 100 ms: E și F activi.

Calculați lățimile benzilor de frecvențe alocate în aceste intervale de timp, vitezele de transmisie și volumul total de date transferat pentru fiecare utilizator.